

N 84-34172

NASA TECHNICAL MEMORANDUM

NASA TM-77758

SYSTEM DESIGN OF A REMOTELY OPERATED, PROGRAM CONTROLLED
MEASUREMENT TABLE FOR THE CALIBRATION OF THE EGRET
EXPERIMENT
PART A: PROGRAM CONTROL

(NONE)

Translation of: Systemauslegung eines fernbedienten,
programmgesteuerten Messtisches für die Eichung des
EGRET-Experiments. Teil A: Programmsteuerung.
(Study carried out under contract of the Bundesministerium fuer
Forschung und Technologie, represented by the Deutsche Forschungs-
und Versuchsanstalt fuer Luft- und Raumfahrt e.V.); ALBEDO GMBH,
Neubiberg (West Germany) and Sondermaschinen Reiter & Co. GMBH,
Munich (West Germany), Report, August 30, 1982, pp. 91-138.

STANDARD TITLE PAGE

1. Report No. NASA-TM-77758	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle SYSTEM DESIGN OF A REMOTELY OPERATED PROGRAM CONTROLLED MEASUREMENT TABLE FOR THE CALLIBRATION OF THE EGRET EXPERIMENT		5. Report Date September 1984	6. Performing Organization Code
		8. Performing Organization Report No.	10. Work Unit No.
7. Author(s) (NONE)		11. Contract or Grant No. NASw 3541	
		13. Type of Report and Period Covered Translation	
9. Performing Organization Name and Address Leo Kanner Associates Redwood City, CA 94063		14. Sponsoring Agency Code	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration, Washington, D.C. 20546			
15. Supplementary Notes Translation of: "Systemauslegung eines fernbedienten, programmgesteuerten Messtisches fuer die Eichung des EGRET-Experiments. Teil A: Programmsteuerung. (Study carried out under contract of the Bundesministerium fuer Forschung und Technologie, represented by the Deutsche Forschungs- und Versuchsanstalt fuer Luft- und Raumfahrt e.V.), ALPEDO GMBH, Munich, (West Germany), Report, August 30, 1982, pp. 91-138.			
16. Abstract Part A of this study concerns particularly program control, it discusses the structure of the program, the five priority levels, the drive routines, the stepwise drive plan, the figure routines, meanders X and Y, the range of measurement table, the optimization of figure drive, the figure drive plan, dialogue routines, stack processing, the drive for the main terminal, the protocol routines, the drive for the microterminal, the drive for the experiment computer and the main program.			
17. Key Words (Selected by Author(s))		18. Distribution Statement Unlimited-Unclassified	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages	22.

SYSTEM DESIGN OF A REMOTELY OPERATED, PROGRAM
CONTROLLED MEASUREMENT TABLE FOR THE CALLIBRATION
OF THE EGRET EXPERIMENTS

PART A: PROGRAM CONTROL

3.01 Structure of the Program

/91*

The microcomputer of program control required a program suitable for its tasks, which should be divided for reasons of clearness in a modular manner into individual partial programs or routines. The functions of the programs must be defined and distributed over these routines in such a way that a minimum in cross relationships is sufficient and errors may be quickly located (Fig. 92-1).

The program must be adjusted to the real time requirements of program control and react within a reasonably short time to external events, whose occurrence is outside its control. The key to the fast reaction processing and the response to external events is represented by the interrupt system of the microcomputer and the possibility for its interface to have direct access to the memory.

In the microcomputer system HF-1000L, the position of an interface decides the priority of its interrupt, that is the possibility of interrupting the courses of programs of lower rank in favor of its own operating routine. With the arrangement of the interfaces according to Fig. 11-1 and the higher interrupt of the processor's own 10ms cycle, this leads to five priority levels:

* Numbers in the margin indicate pagination in the foreign text.

<u>Priority</u>	<u>Interrupt Source</u>	<u>Routines called by the interrupt</u>
1	Internal 10ms cycle	Travel routine
2	Experiment computer interface	Driver routine for the experiment computer
3	Main terminal interface	Driver routine A to C of the main terminal, dialogue routine
4	Microterminal interface	Driver routine of micro-terminal, dialogue routine
5	-	Program background, takes place in the interrupt pauses: Main program, figure routines, protocol, driver routines D and E of the main terminal

The sequence of the priority decreases from 1 to 5. This means for example for the driver routine of the experiment computer, that it can be interrupted at any time for the traveling routine and in its turn for the driver routine of the main terminal interrupt.

/93

Between the routines of different priority levels, a strict separation must exist. They should neither call each other, nor do so jointly with a third party, say a library routine. Violations of this rule would result sooner or later in the collapse of the program because of the loss of the return addresses. The traffic between different levels is for this reason to be carried out

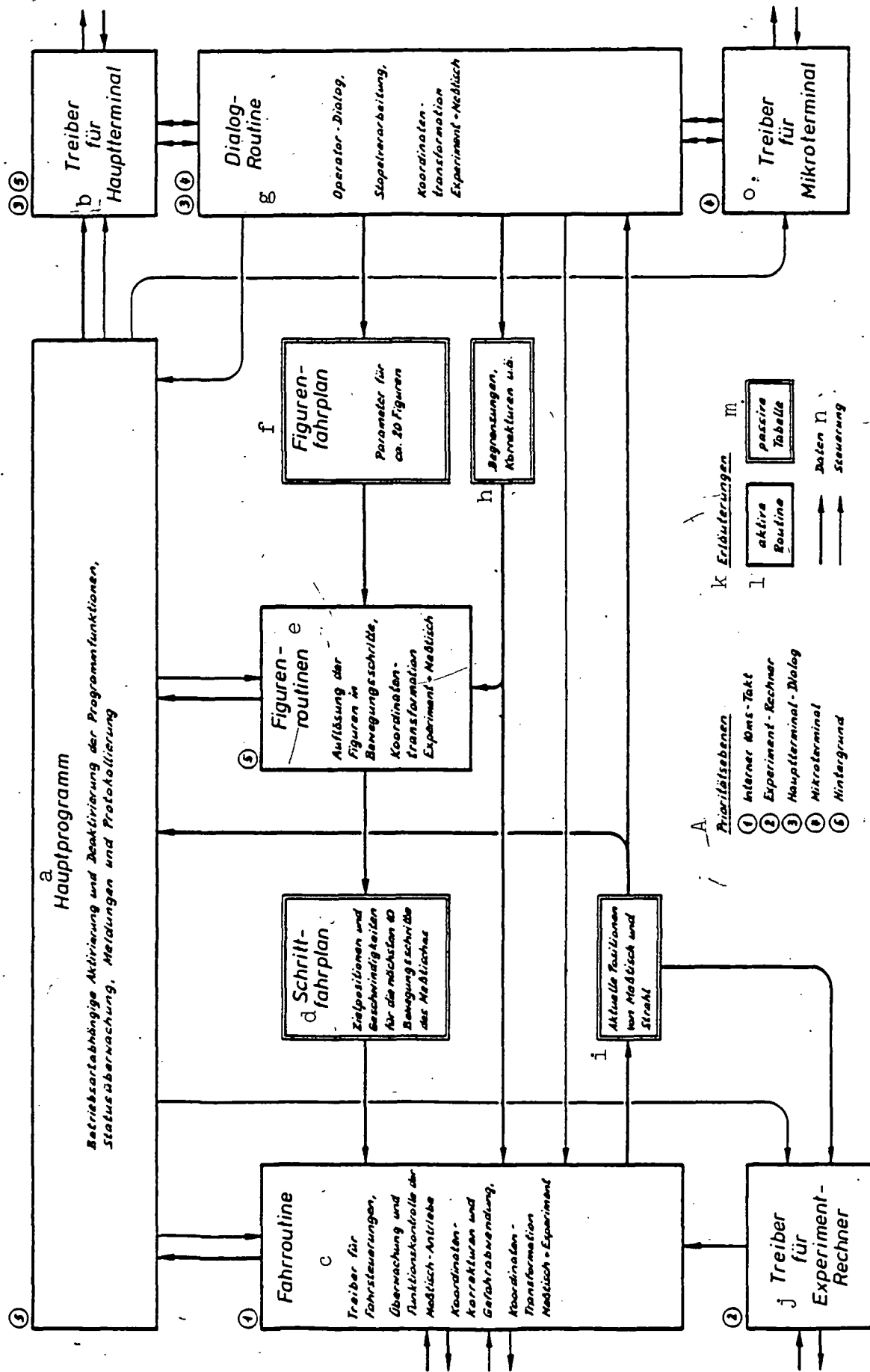


Fig. 92-1 Program Structure
Key to Fig. 92-1 on following page.

Key to Fig. 92-1: A - Priority plan: (1) Internal 10ms cycle; (2) Experiment computer; (3) Main terminal dialogue; (4) Micro-terminal; (5) Background;

- a. Main program, activation and deactivation of the program function depending on the type of operation, status monitoring, announcements and protocol establishment,
- b. Driver for main terminal
- c. Traveling routine, driver for driving control, monitoring and functional control of the measurement table drive, coordinate correction and elimination of danger, coordinate conversion, measurement table experiment
- d. Stepwise traveling plan; target positions and velocities for the next ten stages of movement of the measurement table
- e. Figure routine, resolution of the figures into stages of movement, coordinate transformation, experiment measurement table.
- f. Figure traveling plane, parameters for about 20 figures
- g. Dialogue routine, operator dialogue, stack processing, coordinate conversion, experiment measurement table
- h. Limitation, corrections, etc.
- i. Present position of measurement table and beam
- j. Driver for experiment computer
- k. Discussion
- l. Active routine
- m. Passive table
- n. Data control
- o. Driver for microterminal

through passive markings and tables, while the markings are used predominantly to control the course of the program, while the tables make it possible to make available parameters as data. /94

The same routine which is required at different priority levels, must be copied separately at each of these levels, to avoid the consequences described. Examples of such routines are those for coordinate conversion and for calculating the trigonometric functions.

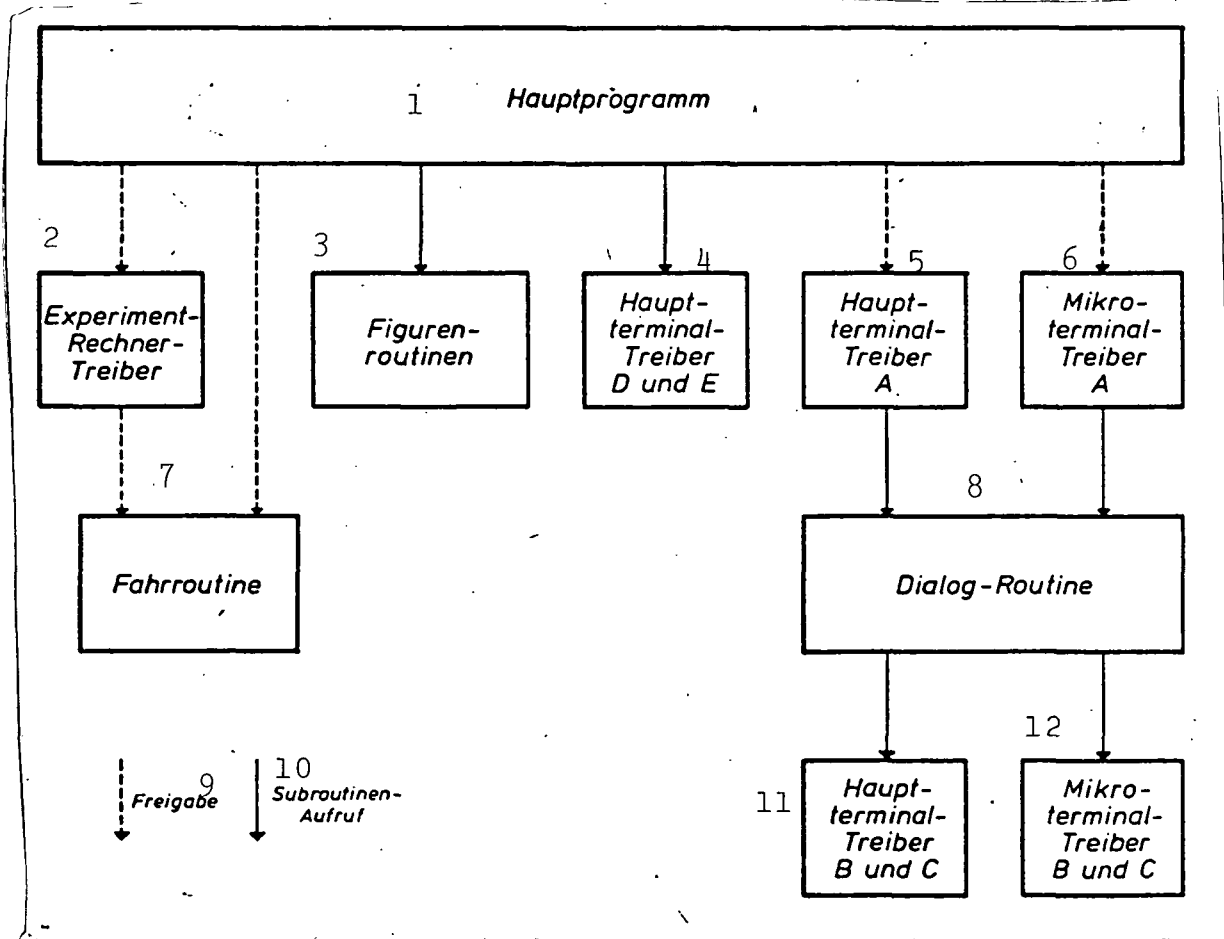


Figure 93-1. Hierarchy of the program routine.

Key: 1. Main program; 2. Experiment computer driver; 3. Figure routines; 4. Main terminal drivers D and E; 5. Main terminal driver A; 6. Microterminal driver A; 7. Traveling routine; 8. Dialogue routine; 9. Release; 10. Subroutine call; 11. Main terminal drivers B and C; 12. Microterminals drivers B and C.

The priority levels decide the sequence in the allocation of processor time to the different functions and routines of the programs. Besides this another order scheme exists in the program, the so-called hierarchy of the program routine, which differentiates between "calling" and "called routines" (Fig. 93-1). The program hierarchy also requires the preservation of a rule, which though trivial, will cause if violated the failure of the program: calling of subroutines should take place always only in one direction, that is the calling routine cannot for its part be called again by the called subroutine.

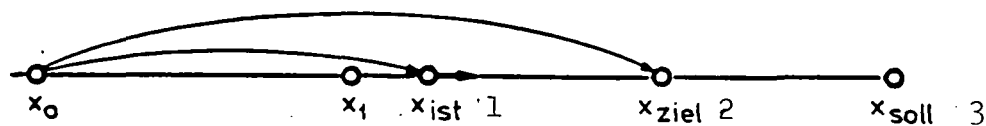
3.02 Traveling Routines

/95

The traveling routine operates and monitors the traveling controls for the four drives of the measurement table. The instructions for it are taken from the stage traveling plan, other operating parameters from the table of the bending corrections and height limitations. The traveling routine is called every ten microseconds by interrupt through a cycle inherent to the processor itself and occupies the highest priority level in the program hierarchy.

Behind the interrupt entrance, the first to be called are the actual nominal values and status data from the four travel controls. Subsequent monitoring sequences test, on the basis of these data, the movement of the operating drives.

We must differentiate between the "approach" and "passage" of a target position x_{ziel} . Whereas in the first case the drive comes to a stop on the target position, in the second case it passes through it with undecreased speed v_{soll} . The two types of behavior are based on the output of different nominal positions x_{soll} for traveling control. For the approach of a target position,



x_0 : Ausgangsposition 4
 x_{ziel} : Zielposition (Vorgabe für Fahrrountine) 5
 x_{soll} : Sollposition (Vorgabe für Fahrsteuerung) 6
 x_{ist} : aktuelle Istposition 7
 x_1 : letzte Istposition 8

Fig.95-1. Stage of movement of a measurement table drive.

Key: 1. Actual; 2. Target; 3. Nominal;
 4. Initial position;
 5. Target position (pregiven for travel routine);
 6. Nominal position (pregiven for travel control);
 7. Actual existing position;
 8. Last existing position

the travel routine assumes the nominal value $x_{soll} = x_{ziel}(\text{target})$, on the other hand for the passage, it selects as nominal x_{soll} one of the two final points of movement x_{min} and x_{max} , which is located in the direction of $x_{ziel}(\text{target})$ (Fig. 95-1):

$$\frac{x_{soll} - x_0}{x_{ziel} - x_0} \geq 1$$

After reaching the target position, recognizable on the status information

$$x_{ist} = x_{soll},$$

the travel routine adjusts the nominal speed value according to the $v_{soll} = 0$ and thereby forces the indifferent behavior of the travel control. The same happens after passing by the target position, that is as soon as we have:

$$\frac{x_{ziel} - x_0}{x_{ist} - x_0} \leq 1,$$

to slow down the drive.

Before reaching the target position, the travel routine currently tests on the basis of the condition

$$\frac{x_{\text{ziel}} - x_0}{x_{\text{ist}} - x_0} > 0,$$

whether the direction of movement is consistent with the position of the target position, and on the basis of a second condition for the stage interval

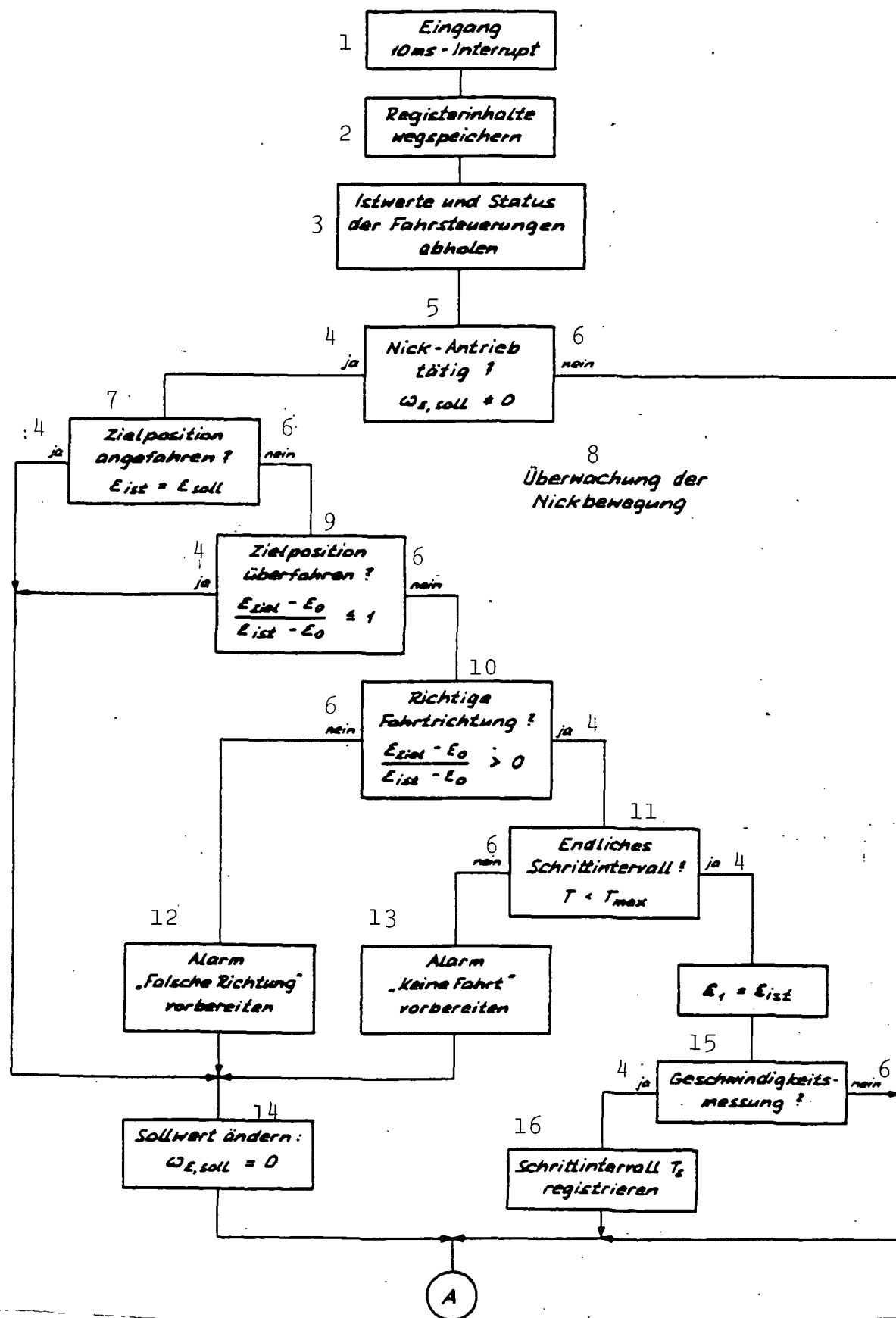
$$T < T_{\text{max}},$$

whether there is travel. Switching errors and effects, which result in the falsification of directions of movement or the failure of a drive, are thus identified and allow the travel routine to stop the drive and sound the alarm.

In case there are no anomalies, the actual present position x_{ist} remains in x_1 for the next passage. Finally to measure the effective travel speed and its possible fluctuation, the state intervals T can be recorded in a sequence.

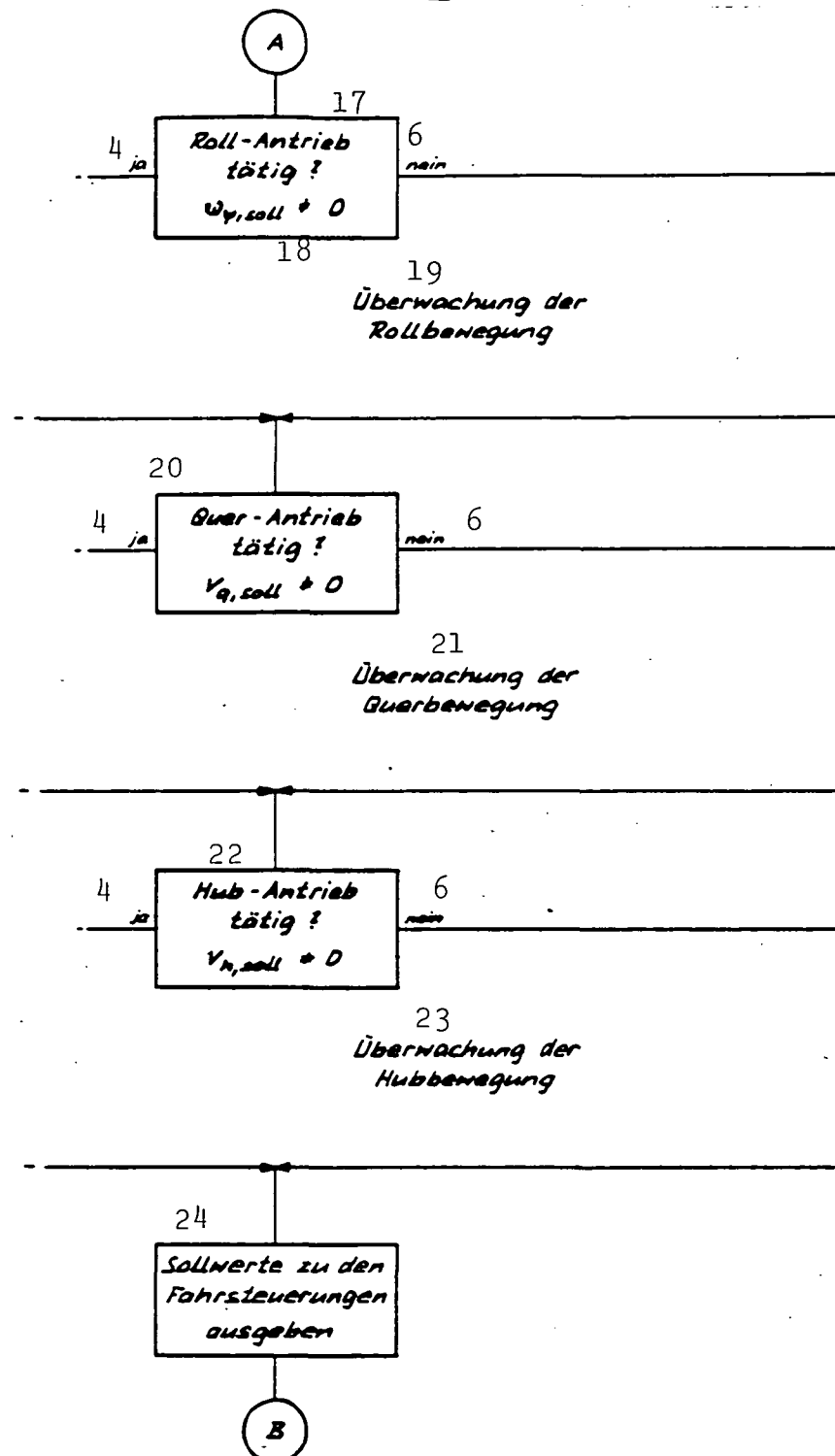
During the activity of the pitching and/or the lifting drive, besides their individual function, they also monitor the preservation of the variable height limitations $h_{\text{unten}}(\text{bottom})$ and $h_{\text{oben}}(\text{top})$ for which horizontal planes of limitation can be programmed below and above the experiment (compare Chapter 2.02). Reaching and exceeding the variable height limitation also causes the stopping of the drives and the release of an alarm. The travel control concludes its activity with the conversion of the nominal values into measurement table and experiment coordinates and the storage of these quantities in the position table. The content of the position table is renewed for every call of the travel routine, that is every ten milliseconds, and independently of the state of activity of the travel controls, and reproduces the actual position of the measurement table or the gamma beam

/97

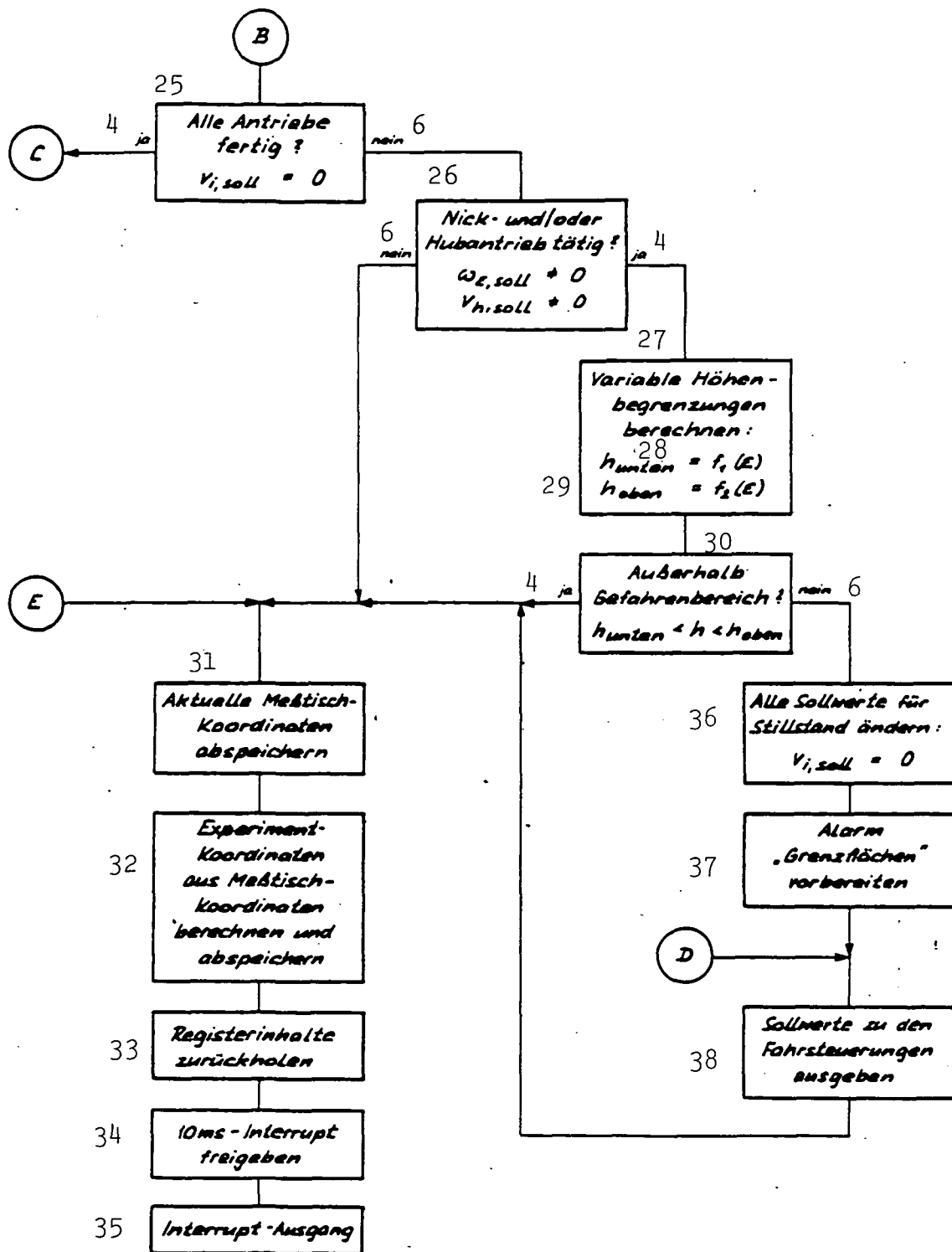


(For key to this figure, see page 13)

TRAVEL ROUTINE

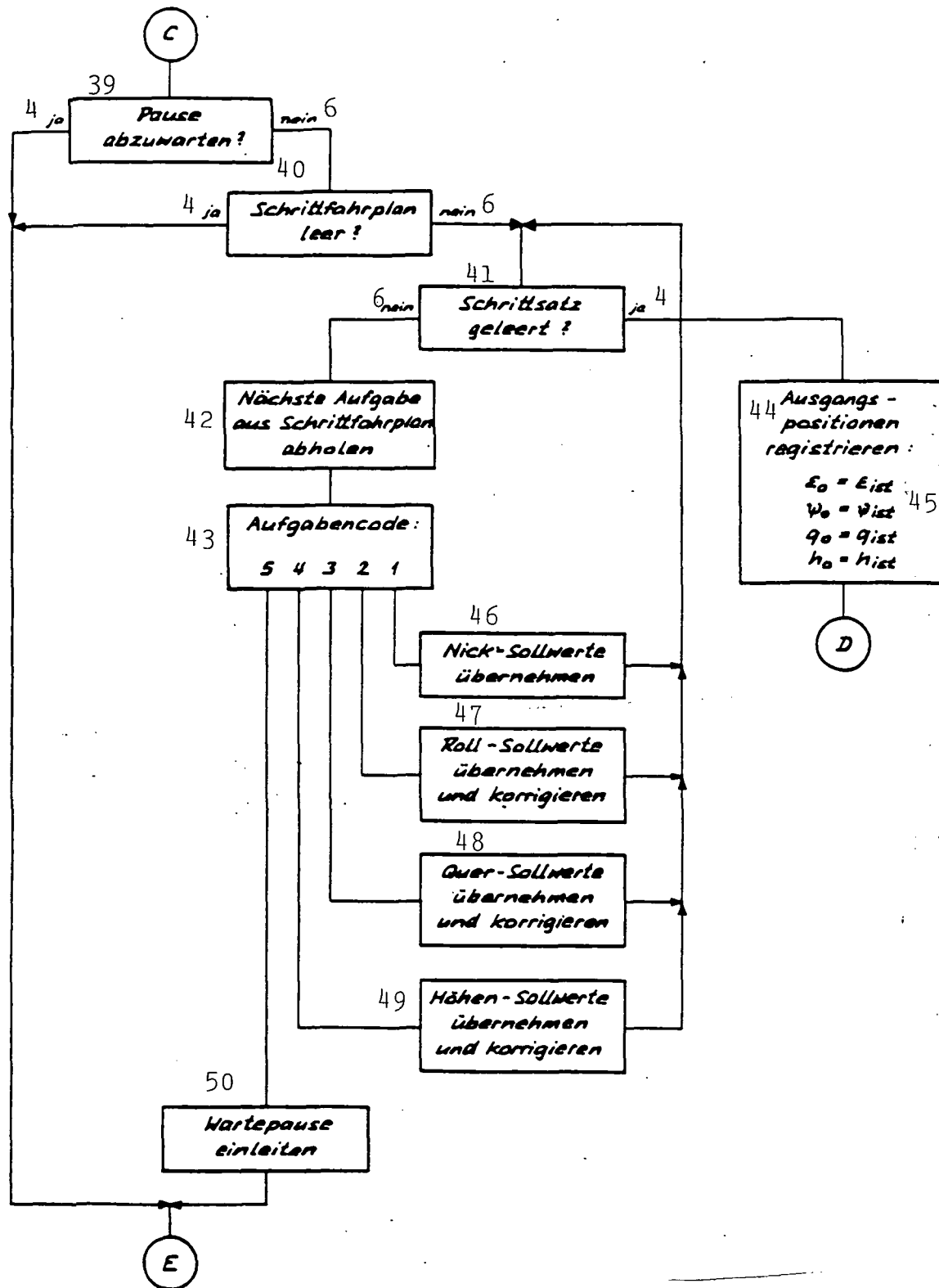


Travel Routine (continued)
(Forkkey to this figure see page I3)



Travel Routine (continued)

(For key to this figure see page 13)



Travel Routine (continued)
(For key to this figure see page 13)

Key to Travel Routine, page 9:

1. Input, 10ms-interrupt; 2. Recording contents stored away;
3. Collection of the actual values and status of travel controls;
4. Yes; 5. Is the pitching drive operating?; 6. No
7. Is the target position reached?; 8. Monitoring of the pitching movement; 9. Is the target position past?; 10. Is the traveling direction correct?; 11. Is it the final stage interval?;
12. Prepare "Wrong Direction" alarm; 13. Prepare "No movement" alarm;
14. Change nominal value; 15. Is the speed measured?;
16. Record state interval T_s ;

Key to Travel Routine (continued) page 10:

17. Is the rolling drive operating; 18. Nominal; 19. Monitoring of the rolling movement; 20. Is the transversal drive operating?;
21. Monitoring of the transversal movement; 22. Is the lifting drive operating?; 23. Monitoring of the lifting movement;
24. Give out the nominal values for drive control;

Key to Travel Routine (continued) page 11:

25. Are all the drives ready?; 26. Is the pitching and/or lifting drive operational?; 27. Calculate variable height limitation;
28. Bottom; 29. Top; 30. Is it outside the danger region?;
31. Store present measurement table coordinates; 32. Calculate and store experiment coordinates from measurement table coordinates;
33. Return register contents; 34. Release 10ms interrupt;
35. Interrupt outlet; 36. Change all nominal values for stoppage;
37. Prepare "Boundary Surface" alarm; 38. Issue nominal values for drive control;

Key to Travel Routine (continued) page 12:

39. Should we wait for a pause; 40. Is the stage traveling plan empty?; 41. Is the stage set empty?; 42. Collect next problem from stage driving plan; 43. Problem code; 44. Record initial positions; 45. Actual; 46. Take over pitching nominal values;
47. Take over rolling nominal values and correct; 48. Take over transversal nominal values and correct; 49. Take over height nominal values and correct; 50. Stop maintenance pause.

even in hand control.

When the last active drive comes to its target position, this causes the end of a stage of movement and causes the travel routine to call upon the next stage set from the stage travel plan. The individual tasks of the stage set are implemented in the manner foreseen, if we are dealing with nominal values, after taking into account the corrections from the bending table for the travel controls.

3.03. Stepwise Travel Plan

/102

The stepwise travel plan is used as a buffer for the transfer of nominal values from figure routines to travel routines. The target positions and nominal velocities calculated by the figure routines for the individual stages of movement are stored in it in sets and are collected in sets by the travel routine as needed, that is after reaching previous target positions.

The advantage of this method is that the travel routine and the figure routines do not generally have to wait for each other, but can carry out their work undisturbed in different priority levels of the program. If the figure routine precisely active is allowed to stop after fulfilling the stage travel plan, and will only become active again when the travel routine has worked out or "emptied" out most of the stage travel plan; then in the pauses between the activity, the processor can be used for other background problems.

Because many stages of movement change only part of the measurement table coordinates and do not require all four drives of the measurement table, it is recommended to establish variable set lengths in the stage travel plan. Such a "stage set" could have the following appearance:

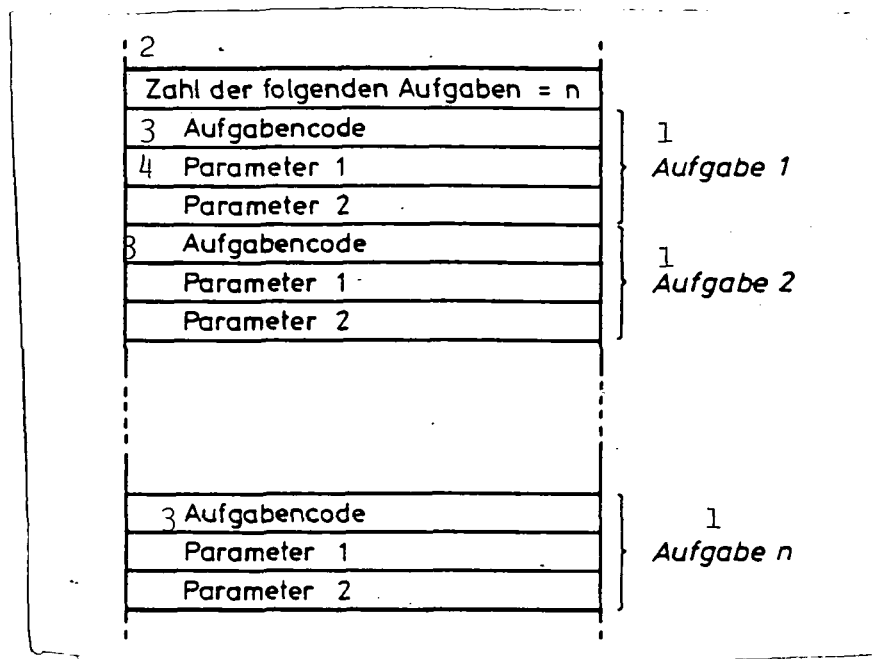


Figure 102-1. Stage set.

Key: 1. Problem; 2. Number of successive problems = n; 3. Problem code; 4. Parameter

Problem codes

1 Pitching

2 Rolling

3 Transversal movement

4 Lifting

5 Illumination pause

. Measurement area

windows, etc.

Drives

Special functions

Parameters

1 Nominal speed, illumination time, etc.

2 Target position, etc.

/103

The problem codes are required because of the variable set formed to differentiate the drives. Besides this they allow the compatibility between figure and travel routine on special functions, for which allowance must be made for the effect on the movement of the measurement table. For example they make it possible to delay the implementation of the next stage of movement until a pregiven duration of illumination has elapsed or the departure is relieved from the outside, say by the experiment computer.

The movement parameters, nominal speed and target position include further instructions to the travel routine. It must be informed, for example, whether the drive should come to a stop on a target position or whether the target position should be passed at the nominal speed.

3.04 Figure Routines

/104

General

The purpose of the figure routine is to calculate a large number of movement stages, in whose implementation by the travel routine a certain movement figure arises, characteristic of the figure routine. The starting point of the calculation is represented by a small number of parameters, which are pregiven by the operator and stored in the figure travel plan.

The parameters describe quantitatively the movement figure and its course. They are, for example, fixed and variable coordinates and screen distances, through which the successive position of the gamma ray beam are determined in direction and position in the reference system of the experiment.

The intermediate result of the routine activity is therefore a sequence of sets of coordinates, which can be described generally in the experiment system by

$$\begin{aligned} & \{ \vartheta_1, \varphi_1, P_1(x_1, y_1, z_1) \}, \\ & \{ \vartheta_2, \varphi_2, P_2(x_2, y_2, z_2) \} = \\ & \quad = \{ \vartheta_1 + \Delta\vartheta_1, \varphi_1 + \Delta\varphi_1, P_2(x_1 + \Delta x_1, y_1 + \Delta y_1, z_1 + \Delta z_1) \}, \\ & \{ \vartheta_3, \varphi_3, P_3(x_3, y_3, z_3) \} = \\ & \quad = \{ \vartheta_2 + \Delta\vartheta_2, \varphi_2 + \Delta\varphi_2, P_3(x_2 + \Delta x_2, y_2 + \Delta y_2, z_2 + \Delta z_2) \}, \\ & \vdots \end{aligned}$$

where of the increments $\Delta\theta_i$, $\Delta\phi_i$, Δx_i , Δy_i , Δz_i , usually only one or very few will always be different from the row.

The coordinates into $\{\theta_i, \phi_i, P_i(x_i, y_i, z_i)\}$ must be converted from the experiment system into the measurement table system (see Chapter 2.01). Of the two solutions

$$\boxed{\{ \varepsilon_{ia}, \varphi_{ia}, q_{ia}, h_{ia} \}} \quad \text{and} \quad \boxed{\{ \varepsilon_{ib}, \varphi_{ib}, q_{ib}, h_{ib} \}}$$

then we have to select the one which can be obtained from the actual position by the shortest means or by any means. It is traveled by the figure routine in the stage travel plan, together with further instructions to the travel routine, which concern for example, the duration of illumination in the individual positions or its ending by the experiment computer.

/105

Meanders in X and Y

As an example we may mention a figure routine "FR1" which meanders over a screen in X and Y with the direction of incidence of the gamma beam remaining the same. The following parameters must be preassigned in the figure travel plan:

Polar angle	(0 to 180.0°)
Azimuth	(-180.0 to 180.0°)
Column distance Δx	(in 0.1 mm)
Line distance Δy	(in 0.1 mm)
Number of columns n	(1 to 100)
Number of lines m	(1 to 100)
Screen height z_R	(in 0.1 mm)
Traveling speed v	(1 to 100%)
Duration of elimination T	(1 to 100s if 0; ending by the experiment computer)

For constant direction of incidence, that is constant polar angle θ and azimuth ϕ , the gamma beam passes consecutively over all the points $P_{ij}(x_j, y_i, z_R)$ of a screen laid in the plane $z=z_R$ with centering around the z axis (Fig. 106-1).

The coordinates of the screen positions P_{ij} may be determined by means of parameters from the figure travel plan

$$\left. \begin{aligned} x_j &= - (n-1) \cdot \frac{\Delta x}{2} + j \cdot \Delta x, \\ y_i &= (m-1) \cdot \frac{\Delta y}{2} - i \cdot \Delta y, \\ z &= z_R \end{aligned} \right\} (F1)$$

The corner in quadrants II assumes for example the coordinates

$$\left. \begin{aligned} x_1 &= -x_{\max} = - (n-1) \cdot \frac{\Delta x}{2}, \\ y_1 &= y_{\max} = (m-1) \cdot \frac{\Delta y}{2}. \end{aligned} \right\}$$

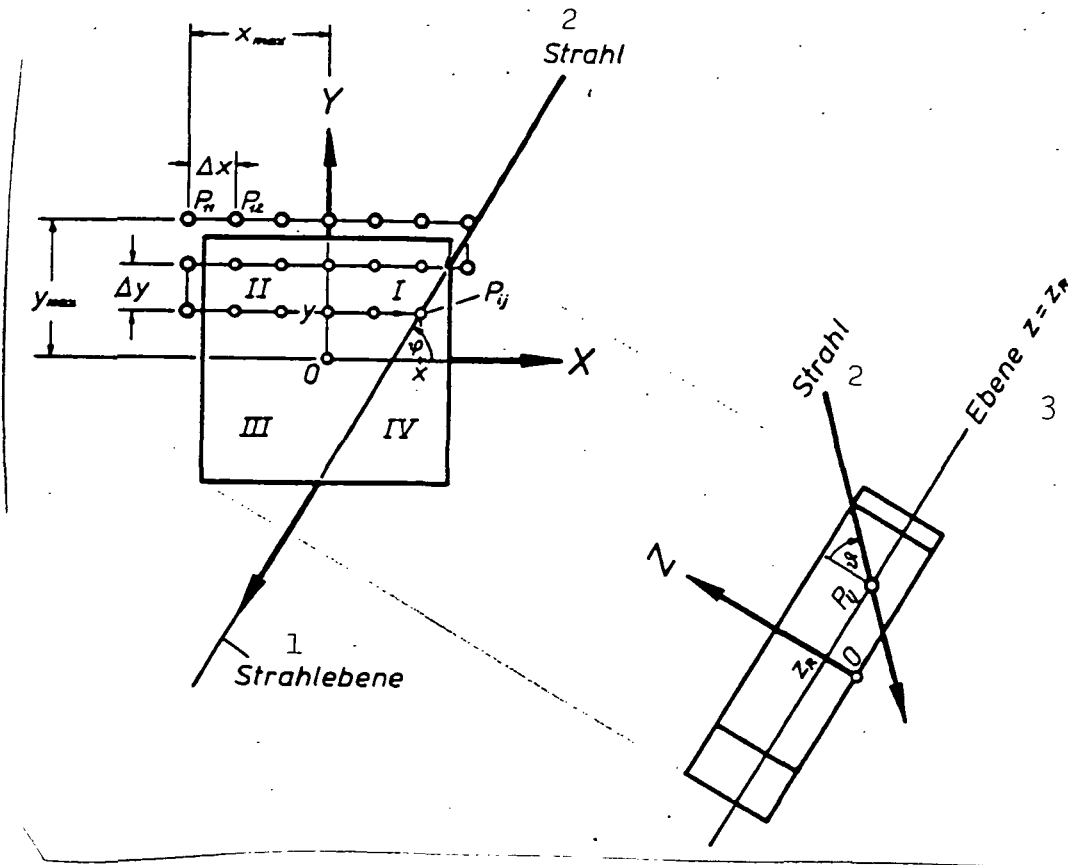


Fig. 106-1: Example of a meander in X and Y
 $(\theta=45^\circ, \phi=60^\circ, m=n=7, \Delta x=\Delta y)$.

Key: 1. Beam plane; 2. Beam 3. Plane

In the conversion from the experiment system to the measurement table system according to Table II (page 60) we obtain as:

Transformation solution A:

Pitch	angle	$\epsilon_a = \theta = \text{const}$	(F2)
Roll	angle	$\psi_a = 270^\circ - \phi = \text{const}$	

Transversal passage	$q_{ija} = x_j \cdot \sin \phi - y_i \cdot \cos \phi =$	(F3)
	$= a_1 \cdot x_j + a_2 \cdot y_i$	
Height	$h_{ija} = x_j \cdot \cos \theta \cdot \cos \phi + y_i \cdot \cos \theta \cdot \sin \phi$	(F3)
	$-(z_R + S) \cdot \sin \theta =$	
	$= b_1 \cdot x_j + b_2 \cdot y_i + b_3$	

and as

Transformation solution B:

Pitch	angle	$\epsilon_b = -\theta = \text{const}$	} (F4)
Roll	angle	$\psi_b = 90^\circ - \phi = \text{const}$	

Transversal passage	$q_{ijb} = -q_{ija}$	} (F5)
Height	$h_{ijb} = -h_{ija}$	

Range of the Measurement Table

/107

Because of the limited clearance of movement and the additional limitations which are required for preventing danger, the measurement table incurs the danger of being unable to reach all the positions of the movement figures which are programmed in the figure travel plan. In so far as the figure routine encounters such position; during the movement, it will bypass it after making a mark in the protocol and will continue with the next screen position which can be reached.

The operator must be left with the possibility open of becoming clear about such effects before carrying out a movement figure. An inherent dialogue part of the figure routine tests therefore, when receiving an order from the operator, the content of the figure travel plan, and announces figure sets with positions which are unreachable, inadmissible parameters and such defects.

Figure 108-1 illustrates the range limitation, which is obtained from the end h_{\min} and h_{\max} of the lifting movement of the measurement table. Here h_{\min} and h_{\max} are naturally to be replaced by the more relevant limit h_{bottom} and h_{top} , if otherwise the horizontal limiting planes could be touched by the experiment (see Chapter 2.02). The figure shows an almost central

position of the gamma beam as compared with h_{\min} and h_{\max} for the example $\theta=50^\circ$ and $\phi=90^\circ$. Solutions of the coordinate conversion are:

Transformation A

$$\begin{aligned}\epsilon_a &= \theta = 50^\circ \\ \psi_a &= 270^\circ - \phi = 180^\circ\end{aligned}$$

Transformation B

$$\begin{aligned}\epsilon_b &= -\theta = -50^\circ \\ \psi_b &= 90^\circ - \phi = 0\end{aligned}$$

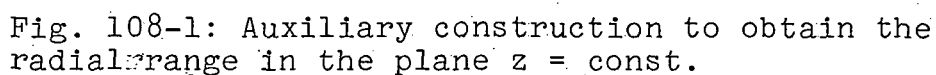
It is found that the conversion solutions A and B offer ranges of equal size (y_{\max}) in the direction of the positive y axis and much smaller ones (y_{\min}) in the opposite direction. Both conversions cover therefore almost overlapping regions which are inadequate in the direction of the negative y-axis. For the range of the measurement table, overlapping is a drawback, since it is impossible to see when a position can be traveled more than once.

The results are improved with the increasing distance of the beam course from the central position between h_{\min} and h_{\max} . For example in Fig. 108-1, with the further downward displacement of the beam the axis sections $y_{\max}(A)$ and $y_{\min}(B)$ increase at the cost of $y_{\max}(B)$ and $y_{\min}(A)$. The mutual overlapping of the regions which may be reached with the conversion A and B decreases, the surface which can be reached altogether by A or B increases.

/108

By means of Table 1 on the coordinate conversion (page 59) we obtained the length of the axis sections after substituting s by $s + z_R$ in (9):

$$\left. \begin{aligned}y_{\min}(A) &= \frac{h_{\min}}{\cos \epsilon} + (s + z_R) \cdot \operatorname{tg} \epsilon \\ y_{\max}(A) &= \frac{h_{\max}}{\cos \epsilon} + (s + z_R) \cdot \operatorname{tg} \epsilon\end{aligned} \right\} \quad (F6)$$



/109

The sum of the axis sections is naturally the same for both conversions, independently of the beam height and is only a function of the lifting range $h_{\max} - h_{\min}$ and the pitching angle ϵ :

22

The horizontal movement clearance of the measurement table extends between its ends q_{\min} and q_{\max} . By adjusting the measurement table with regard to the course of the gamma beam, an attempt is made to achieve a central position, that is $q_{\min} = -q_{\max}$. Then the experiment can have its z axis equally broad on both sides, specifically at a distance of $\pm q_{\max}$ from the beam and should travel completely away from the beam.

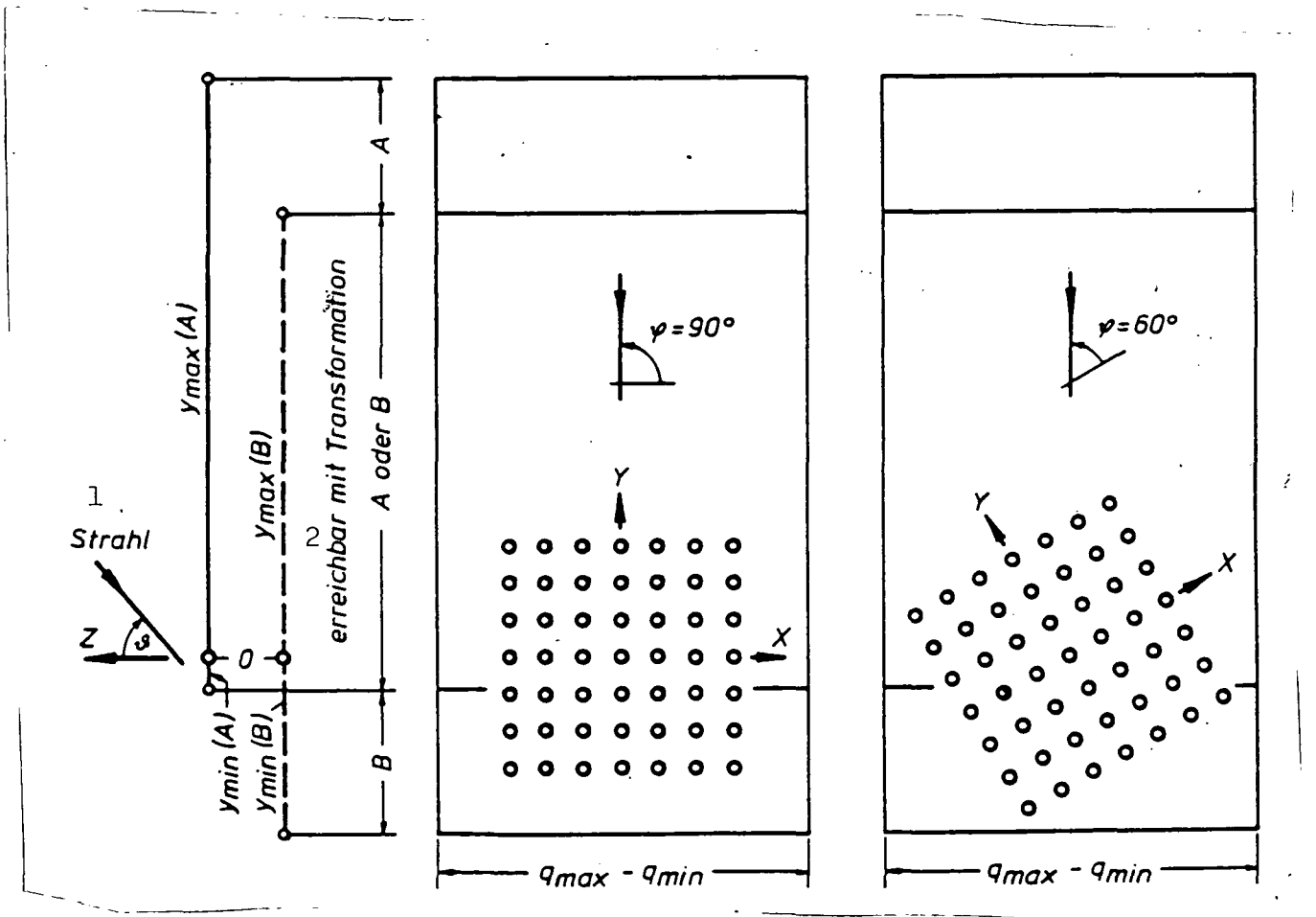


Fig. 109-1: Radial range limitation in a plane $z = \text{const}$ as a result of final transversal and height adjustments.

Key: 1. Beam; 2. Achievable with conversion A or B

Figure 109-1 shows the region of the plane $z=z_R$, which can be reached by the beam by height and transversal adjustment of the measurement table and by means of conversion A and B alone or overlapping. The conditions of Fig. 108-1 were taken for the representation and shown on the left edge. The screen positions of a meander are represented for the azimuth angle $\phi=90^\circ$ and 60° and are altogether in the conversion region B, but only partly in the region of conversion A.

In shifting the beam course in the center between h_{\min} and h_{\max} , the regions A and B were contracted until they overlapped completely, with the result that points in the lower portion of the screen could no longer be reached by the beam.

Whether a screen point $P_{ij}(x_j, y_i, z_R)$ lies in the range of the measurement table may be established numerically by means of the following equations of conditions in which the transversal displacement q_{ij} and height h_{ij} from (F3) and (F5) should be introduced:

Conversion A

$$\begin{aligned} q_{\min} &\leq q_{ija} \leq q_{\max} \\ h_{\min} &\leq h_{ija} \leq h_{\max} \\ h_{\text{bottom},a} &< h_{ija} < h_{\text{top},a} \end{aligned}$$

Conversion B

$$\begin{aligned} q_{\min} &\leq q_{ijb} \leq q_{\max} \\ h_{\min} &\leq h_{ijb} \leq h_{\max} \\ h_{\text{bottom},b} &< h_{ijb} < h_{\text{top},b} \end{aligned}$$

The variable height limitation h_{bottom} and h_{top} prevent the experiment from approaching the structures of the measurement table and its vicinity. Chapter 2.02 discusses it and its dependence on the pitching angle ϵ .

Optimization of the Figure Travel

The calculation and conversion of the coordinates for the successive positions takes place against the background of the program course, that is with minimum priority. The figure routine combines all the data for implementing a movement stage, that is target position, nominal speed and other instructions to a stage set and stores the latter in the stage travel plan. After fulfilling the stage travel plan the figure routines must be able to interrupt their work without loss of the intermediate results, so that the other background functions, for example protocol guiding should have their chance. As soon as a supply in the stage travel plan has combined into a last stage set, the figure routine resumes its work once again. /111

In the conversion of the screen positions, the figure routines must decide which of the two solutions A and B should be preferred each time. The purpose of the decision is to reduce the travel part and times of the measurement table to the minimum. Because the coordinate changes of the same movement stage take place usually at the same time, the time Δt needed for this stage of movement is determined only by the duration of the longest coordinate change:

$$\Delta t = \text{Max} \left(\left| \frac{\Delta \epsilon}{\omega_{\epsilon}} \right|, \left| \frac{\Delta \psi}{\omega_{\psi}} \right|, \left| \frac{\Delta q}{v_q} \right|, \left| \frac{\Delta h}{v_h} \right| \right).$$

Strictly speaking the travel times Δt of a movement figure should not be optimized individually, but also as a sum. Also by varying the sequence in which the screen positions can be covered, in most cases time advantages can be gained. Nevertheless a detailed preliminary planning of the movement figure does not seem to be reasonable. It would require figure routines, whose complexity and cost would not always be proportionate to the benefit.

Giving of generalization and consideration of a special movement figure, in most cases rules and criteria simple to apply could lead travel time optimization. As an example we will once again give an XY-meander with constant polar angle and azimuth, and specifically for fixed and variable sequence of the screen positions. The main object is to avoid unnecessary change between conversion areas A and B during the figure travel, probably because these require each time a change of the pitch position by the amount $|\Delta\varepsilon|=2\theta$ and the roll angle by $|\Delta\psi|=180^\circ$. Let us assume for the study that the position of the screen is at the overlapping boundary of region A and B.

Fixed Meander Path

/112

The sequence of screen position with initial and final point is preassigned in a fixed manner. The figure routine can only select freely for each of the positions the conversion solution A or B. To clarify our considerations, in the following screen with 4×4 positions, it is assumed that their converted coordinates may lie outside the reachable regions A and B (empty circles):

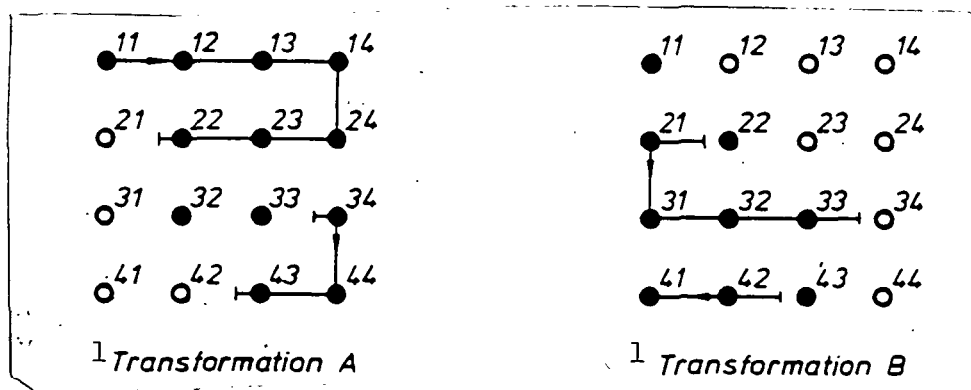


Figure 112-1

Key: 1. Conversion

On the path indicated the figure routine meets three time coordinates which can only be reached after changing the conversion.

As we see in point 11 we could also begin with conversion B, but with the consequence of a further change of conversion before point 12. The considerations lead to the following simple behavior rules:

1. Beginning of the meander path with the conversion which can be maintained over the largest number of subsequent positions;
2. Change of conversion only when the following position cannot be achieved by any other means.

Rule 1 is only applicable if position 11 corresponds to both conversion regions, and requires a preliminary exploration of the screen different from rule 2. But in this connection it is sufficient to study only four corner positions representing the entire screen. The difference sequence of the corner positions for even and odd number of lines m should not be overlooked here:

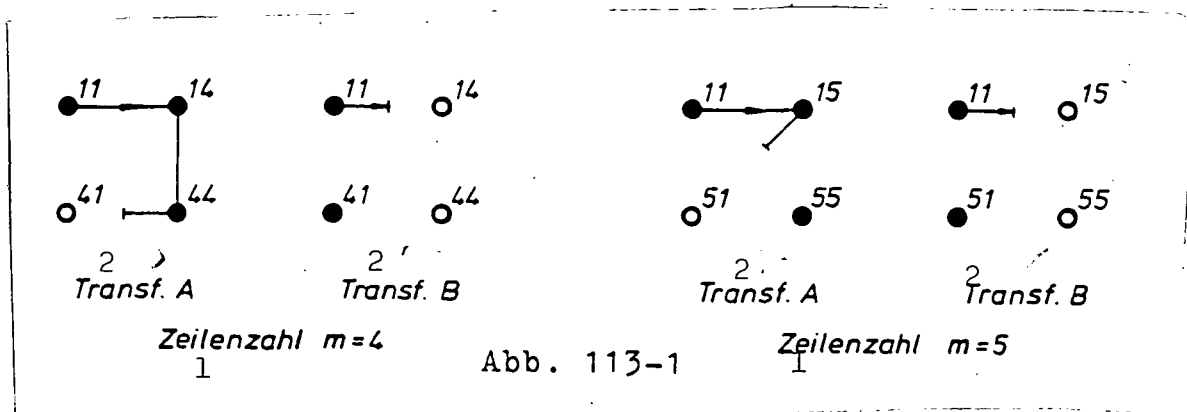


Fig. 113-1

Key: 1. Number of lines; 2. Conversion

Variable Meandering Path

The figure routine is free to begin the meander in any of the four corner positions and to proceed by line or column. The selection criterion is again avoiding as far as possible changes of conversion. The study of the above 4 x 4 screen shows that it is possible to manage with a single change with variable meander path:

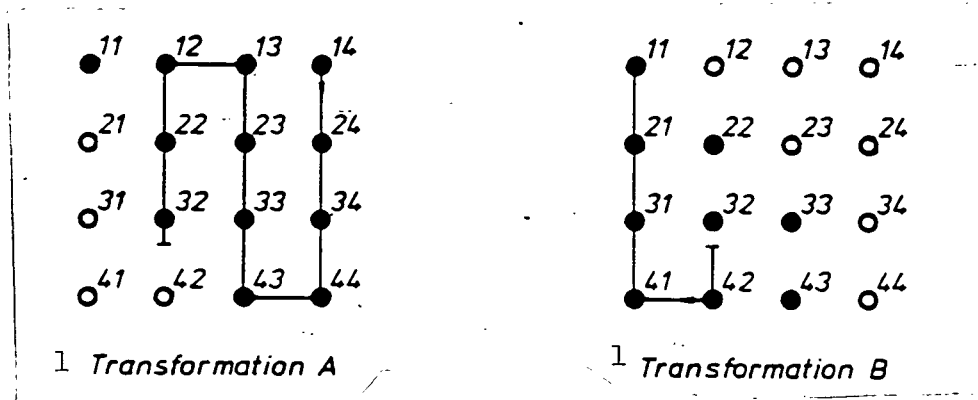


Fig. 113-2

Key: 1. Conversion

Behavior rules can be found once again empirically:

1. Counting of the lines and columns, which can be covered without changing the conversion. When traveling in the direction of lines, if the number of lines covered predominates, otherwise travel in column direction.

2. Beginning of the meander path in any corner position and in such a way that a maximum number of positions can be covered before the first change of conversion.

3. Change of conversion only when the following positions cannot be reached otherwise.

Rules 1 and 2 require a preliminary exploration of the screen, for which naturally only the corner position need be studied.

Because of the linear limits of the conversion regions A and B, 114 it may be assumed moreover to simplify matters that all positions of a line or column belong to one region, if this is proven for the two boundary positions of this line or column:

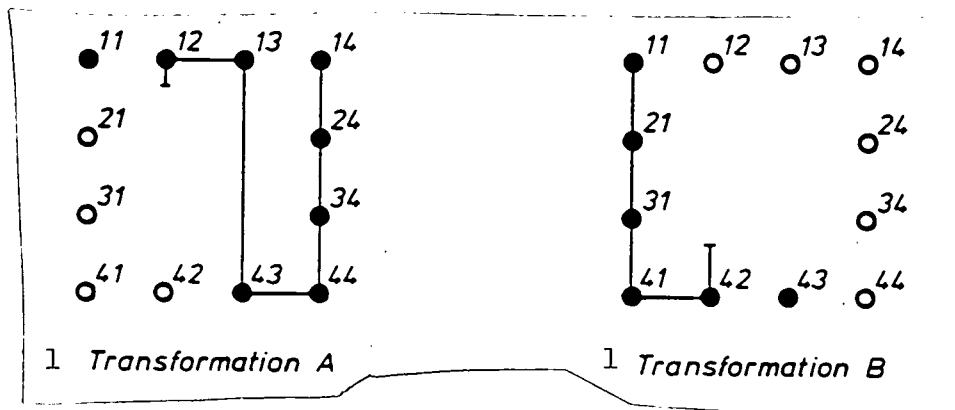


Figure 114-1

Key: 1. Conversion

Free Path

The figure routine always achieves a single change of conversion, when it allows the coverage first of all the screen positions in one region and then the remaining in the other region. Naturally account must be taken of the positions already covered:

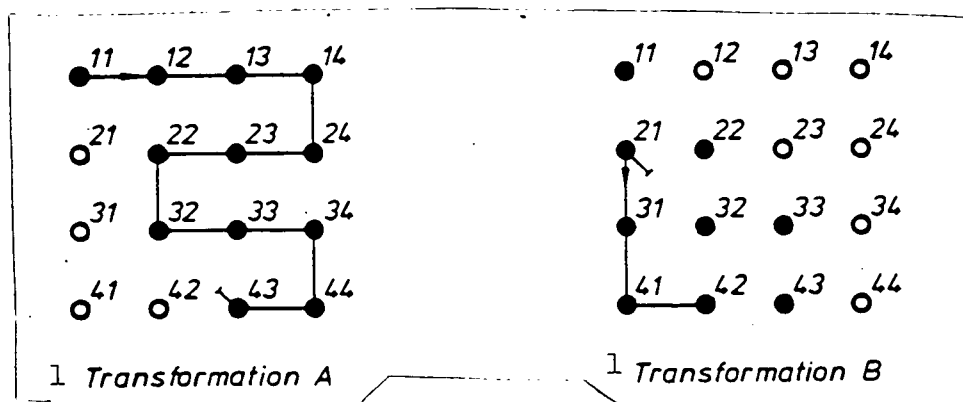
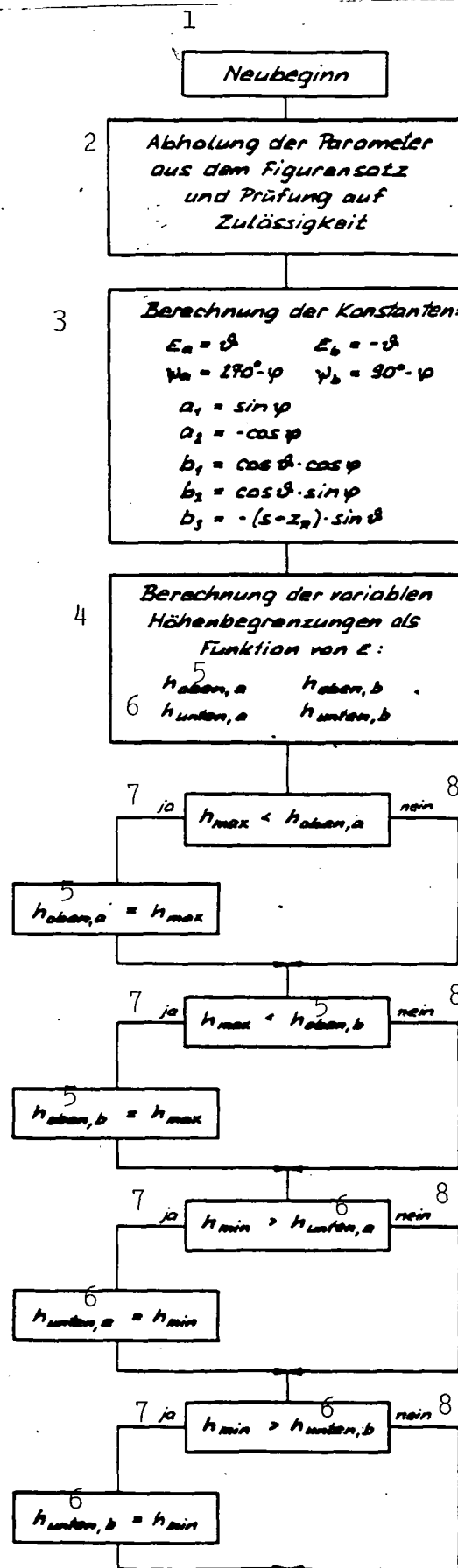


Fig. 114-2.

Key: 1. Conversion



(Key to this figure on page 34) A

Figure Routine "Meander in X and Y"

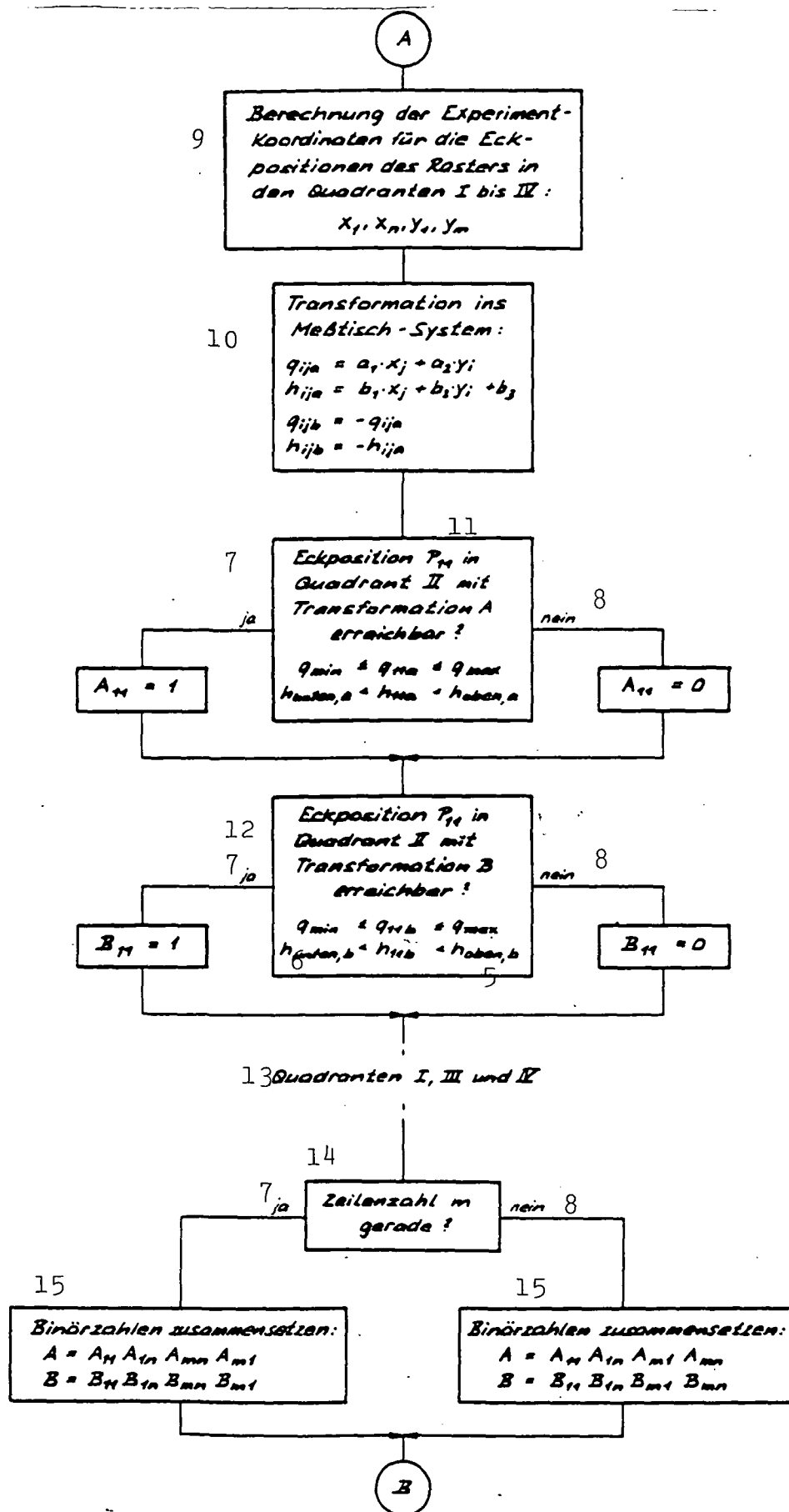
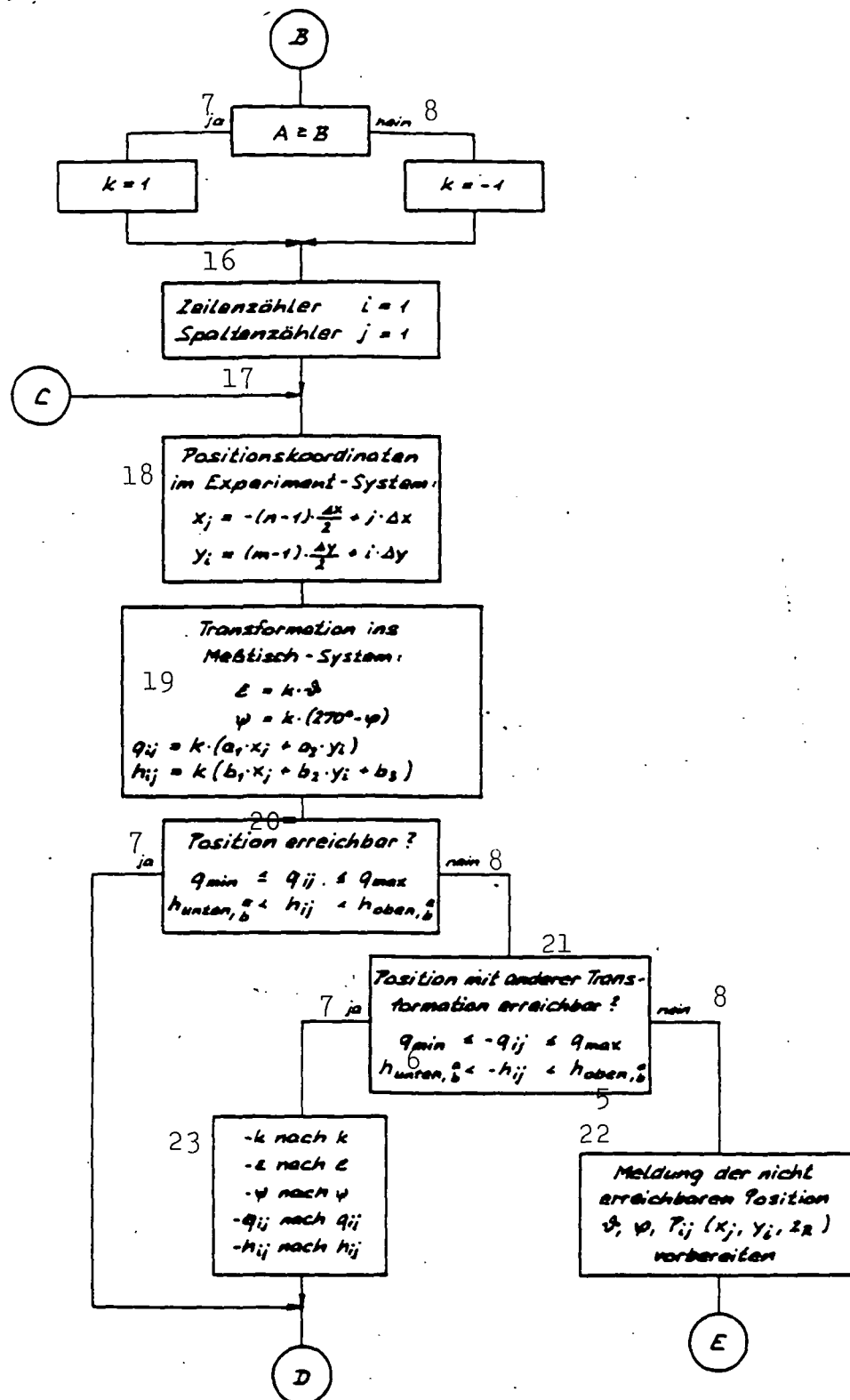


Figure Routine (continued)
(for key to this figure see page 34).



(for key to this figure see page 34).

Figure Routine (continued)

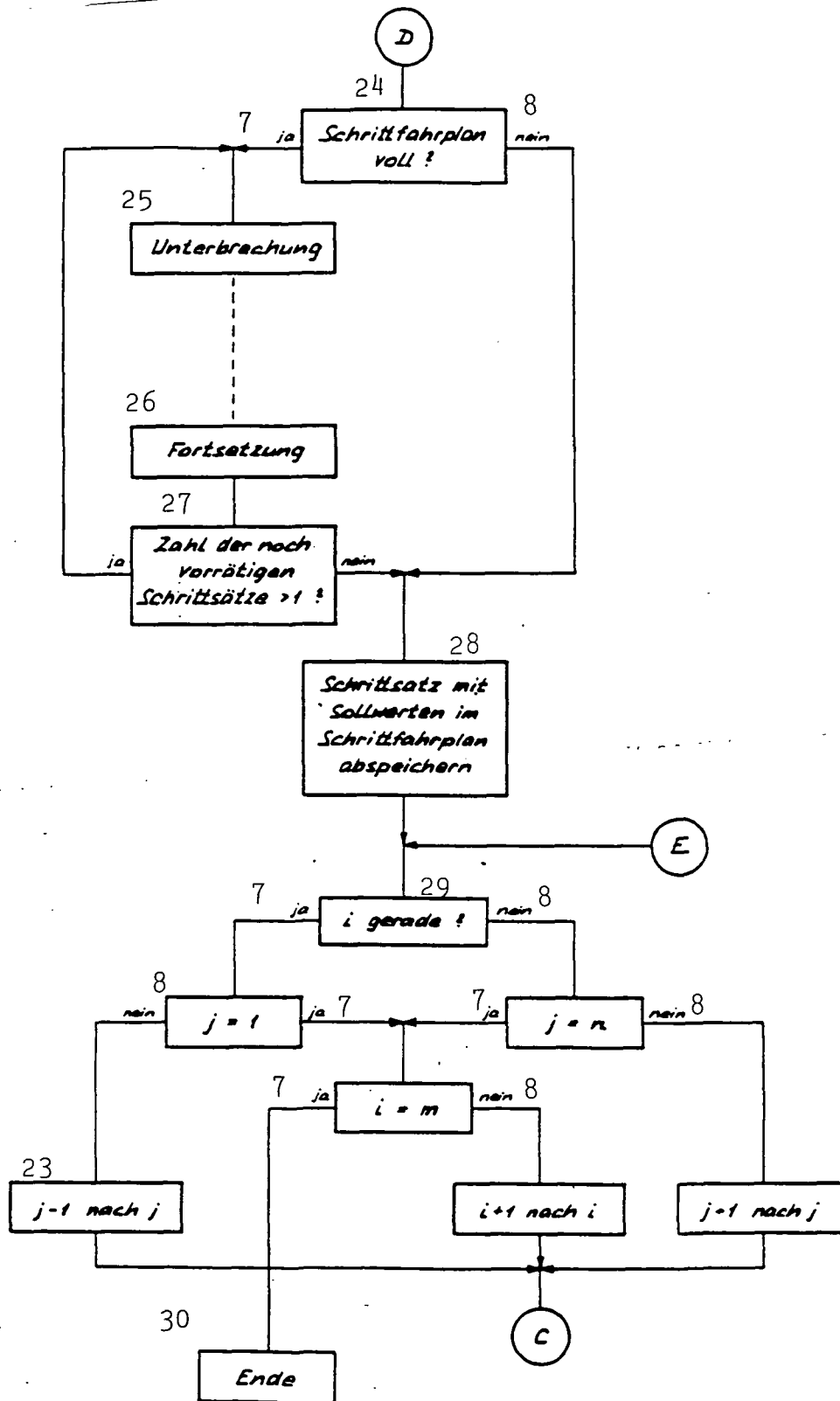


Figure Routine "Meander in X and Y" (continued)

(For key to this figure see page 34)

Key to Figure Routine "Meander in X and Y", page 30:

1. New beginning; 2. Collection of the parameters from the figure set and testing of the reliability; 3 Calculation of constant;
4. Calculation of the variable height limitation of the function of ϵ ;
5. Top; 6. Bottom; 7. Yes; 8. No;

Key to Figure Routine "Meander in X and Y" (continued) page 31:

9. Calculation of the experiment coordinates for the corner positions of the screen in quadrants I to IV;
10. Transformation into the measurement table system;
11. Can corner position P_{11} in quadrant II be reached with conversion A?;
12. Can corner position P_{11} in quadrant II be reached with conversion B?;
13. Quadrants I, III and IV; 14. Is the number of lines m even?;
15. Consistent binary numbers;

Key to Figure Routine "Meander in X and Y" (continued) page 32:

16. Line counter; 17. Column counter; 18. Position coordinate in the experiment system; 19. Conversion into the measurement table system; 20. Can the position be reached?;
21. Can the position be reached with a different conversion?;
22. Prepare the announcement of the unreachable position $\theta, \phi, P_{ij}, (x_j, y_i, z_R)$; 23. Towards;

Key to Figure Routine "Meander in X and Y" (continued) page 33:

24. Is the stage travel plan full?; 25. Interruption;
26. Continuation; 27. Number of the stage set still in stop?;
28. Store stage set with nominal values in stage travel plan;
29. Even; 30 End.

The dialogue orders of program control (Chapter 2.05) may be divided into three types "note", "continue until further order" and "do now." Examples of this are:

FAN - Connection of figure sets

BFF - Conversion "figure travel according to travel plan"

SEP - Control: Positions experiment

The first two types and the activities prepared and in progress are those which actually justify the designation "program control." They allow the operator to program the program control, to feed it therefore, a program of future courses and to call for it at a given time.

To prevent possible misunderstandings: here the much used term "programming" is not intended to refer to the programming of the microcomputer, but to the input of tabulated parameter sequences taking place per dialogue or from the magnetic tape, and whose purpose is to establish the movement figures of the measurement table to be carried out in the future.

The program section fed by the operator in the form of individual figure sets close in themselves is deposited by the program control in the figure travel plan and fills out there areas different in number. The "F" orders of the dialogue allow the operator to feed and establish the figure sets at the terminal, their activation and deactivation and the transfer of the figure travel plan to and from the magnetic tape cassettes.

The feed of a figure set from the terminal and its storage in the figure travel plan may be represented on the basis of an example (compare "Meander in X and Y", Chapter 3.04):

FTE	Dialogue order: "Feed figure set from terminal"	
F9	Field number of the figure travel plan: Figure set 9	
1	Switching sign: 1 - figure set active Ø - figure set inactive	
FR1	Designation of figure routine	<u>/120</u>
"XY-MEANDER"	Announcement text, appears in protocol This is followed by series of parameters, whose significance is established by the selected figure routine, here FR1:	
3ØØ	P1 - Angle $\theta=30.0^\circ$	
-45Ø	P2 - Azimuth $\phi=-45.0^\circ$	
1ØØ	P3 - Column distance $\Delta x=10.0$ mm	
1ØØ	P4 - Line distance $\Delta y=10.0$ mm	
2ØØ	P5 - Column number $n=200$	
2ØØ	P6 - Number of lines $m=200$	
Ø	P7 - Screen height $z_R \approx 0$	
1ØØ	P8 - Traveling speed $v=100\%$	
3ØØ	P9 - Duration of lighting $T=30.0$ s	
E	End	

After storage the field 9 of the figure travel plan offers
the following picture:

Among all program routines the dialogue routine is the most extensive, because it must differentiate all permissible order patterns and convert the wishes of the operator into the corresponding measures and activities of program control. Because of the isolation in it of the individual priority planes, it must cover the traffic with other program routines exclusively through parameter tables, such as for example the figure travel plan, or by placing marks (compare Chapter 3.01).

After calling the dialogue routine through the driver A of the main or microterminal, the dialogue continues with the operator with the help of the driver routine B, first until the completion of the first feed set, which usually contains an order call. If it is a permissible order pattern, the dialogue routine is converted into a sequence specially suitable to implement this order:

After a "B" order for conversion of the type of operation (see Chapter 2.05) through the dialogue routine markings are placed in the main program which are interpreted later by the main program, and which also imply the activation and deactivation of drivers and other program functions.

"F" orders cause the dialogue routine to have access to the figure travel plan and there carry out the changes desired by the operator.

"M" orders are aimed at the transfer of data from or to the cassette running gear of the main terminal or its control, and are carried out by means of driver routines B and C.

Finally "S" orders are differentiated according to whether the actual coordinates of measurement table or experiment must be indicated or recorded, or whether the travel routine must be

carried out for a movement of the measurement table. In the first case the dialogue routine has to be operated only from the table of actual position data. The second case imposes higher requirements, since the travel routine must give nominal values for the movement stage to be carried out. Moreover another conversion must be carried out from the experiment to the measurement table system, if the order of the operator referred to the experiment.

/122

If answer sets must be issued to the operator, the dialogue routine uses driver routine C of the corresponding terminal, regardless of whether we are dealing with information, instructions or auxiliary statements.

Stack Processing

In stack processing prepared order and parameter sets must be read from a magnetic tape cassette and carried out or announced in the same way by program control as though it were said to the operator in dialogue. The type of operation "figure travel according to stack" changes for this purpose the behavior of the dialogue routine: whereas the latter waits otherwise for the operator's initiative, here it reads itself or with the help of the main program the next instruction from the tape, as soon as the previous activity of program control has come to an end.

3.07 Driver for the Main Terminal

/123

To conduct the dialogue between main terminal and program control and to give out the protocol to the main terminal, a series of driver routines are needed. These drivers belong to two different priority levels and operate either on an interrupt or contain an interrupt pause in their sequence. In case of input to the dialogue, the drivers test and store the incoming symbols individually. All data output on the other hand takes place in direct memory access (DMA).

Dialogue Drive A

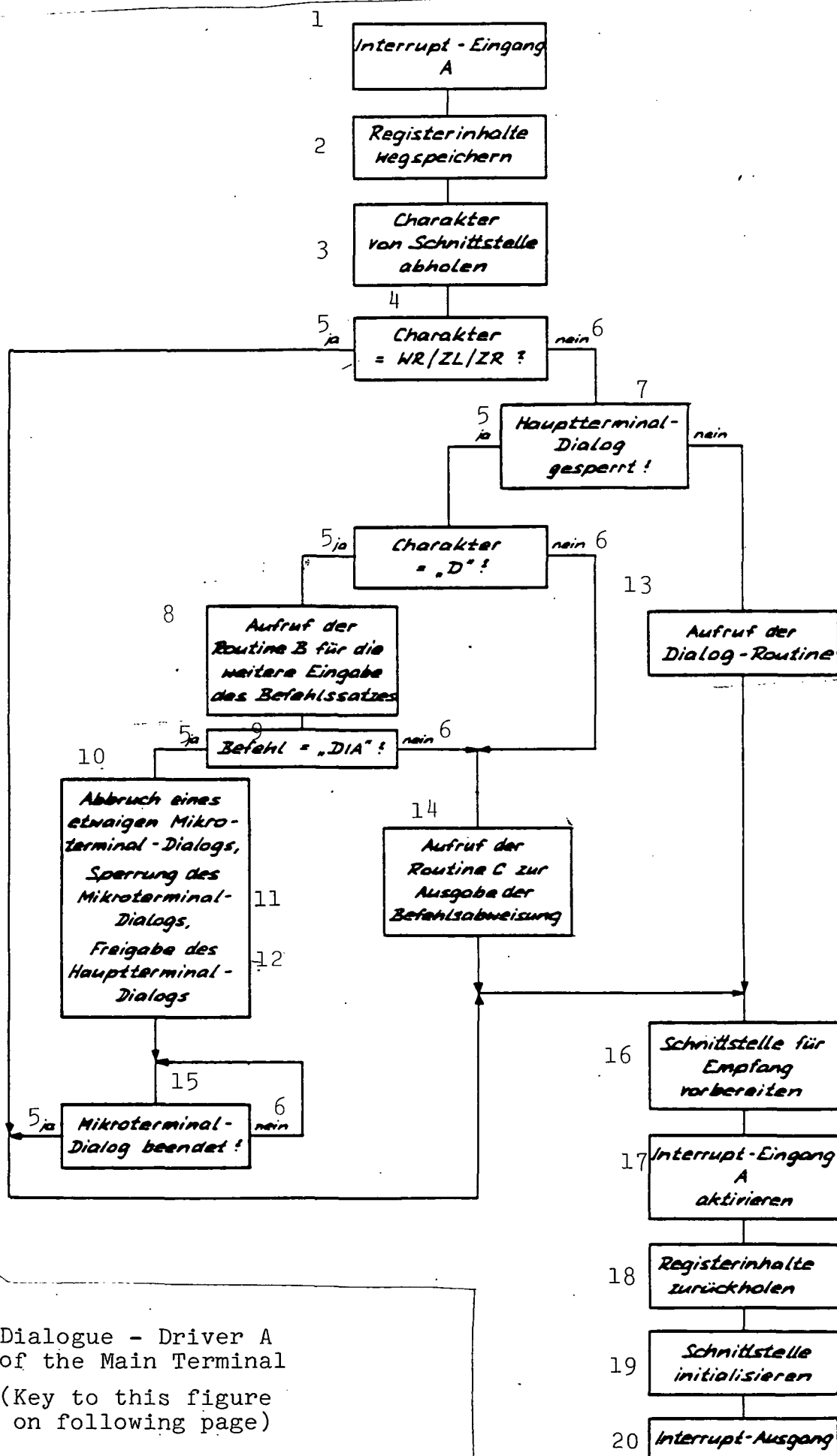
The interrupt input A is always activated in the activity pauses of the main terminal, that is when the latter is not occupied either by the dialogue or by the protocol. The scanning field of the main terminal and the interface of the program control are released simultaneously for input or reception. The stop of a key in this phase means with the exceptions given below an attempt to enter the dialogue with program control.

The interrupt routine A tested by pressing the key studies first whether the symbol fed concerns one of the symbols for the carriage return (WR), line feed (ZL) or gap (ER). The latter should make it possible to obtain the distance on the display screen of the terminal, and are covered by routine A without further consequences. Any other symbol is considered as an attempt to take up the dialogue and leads when the main terminal dialogue is released to the activation of the dialogue routine.

Otherwise, if the main terminal is blocked, the operator must be allowed to feed the switching order "DIA" (compare Chapter 2.05). Thus for the symbol just fed around "D," the operator by calling the driver routine B has the opportunity to complete the order set which was begun. If subsequently "IA" stands in the text buffer, routine A allows interruption and blocking of the microterminal dialogue and the release of the dialogue at the main terminal. All other input sets different from "DIA" are rejected.

/125

After return from the dialogue routine or after rejection of a block dialogue order, the driver routine A ends. Beforehand it also connects the interface to reception and activates the interrupt input A, therefore restores the state which had been present at the beginning of its activity.



Key to Figure Dialogue - Driver A of the Main Terminal:

1. Interrupt input A; 2. Store away register content;
3. Collect symbols from interface; 4. Symbol; 5. Yes; 6. No;
7. Main terminal dialogue blocked!; 8. Call of routine B for
further input of order sets; 9. Is the order "DIA"?;
10. Interruption of a possible microterminal dialogue;
11. Blocking of the microterminal dialogue; 12. Release of the
main terminal dialogue; 13. Call of the dialogue routine;
14. Call of routine C for issuing order rejection;
15. Is the microterminal dialogue ended?; 16. Prepare interface
for reception; 17. Activate interrupt input A; 18. Return register
content; 19. Initiate interface; 20. Interrupt outlet

Dialogue Driver B

Driver B is called by dialogue routine when the order and parameter sets are to be received and stored in a text buffer. After erasing the text buffer the driver routine B prepares the interface, the scanning field of the main terminal and the interrupt input B for the feed of the individual symbols. The interrupt pause, which is the time from the initiation of the interface until the encounter with a symbol, makes it available for carrying out problems of subordinate priority planes of the program.

After the arrival of a symbol by comparison with possible patterns, it is tested whether it must be treated in a special way or only in the text buffer; the symbols for carriage return (WR) and gap (ZR) are passed. The line feed (ZL) or comma (,) apply as the end of the input set and cause routine B to transfer the text received to dialogue routine. The return stage symbol (RS) closes the latest of the symbols stored in the text buffer to be erased. Finally the erase symbol leads to the interruption of the activity of routine B and further to the dialogue routine. All other symbols are collected in the sequence of their occurrence in the text buffer. Excess symbols, which can not find any place in the text buffer, are lost.

Routine B repeats the initiation of the interface and its waiting until the next symbol, until the input set has come to its proper end (comma, line shift) or is interrupted (erasure). Instead of the erasing symbol on the main terminal, the activity of routine B can also be interrupted by the switching order "DIA" from the microterminal and specifically by means of this driver routine A. /127

Dialogue Driver C

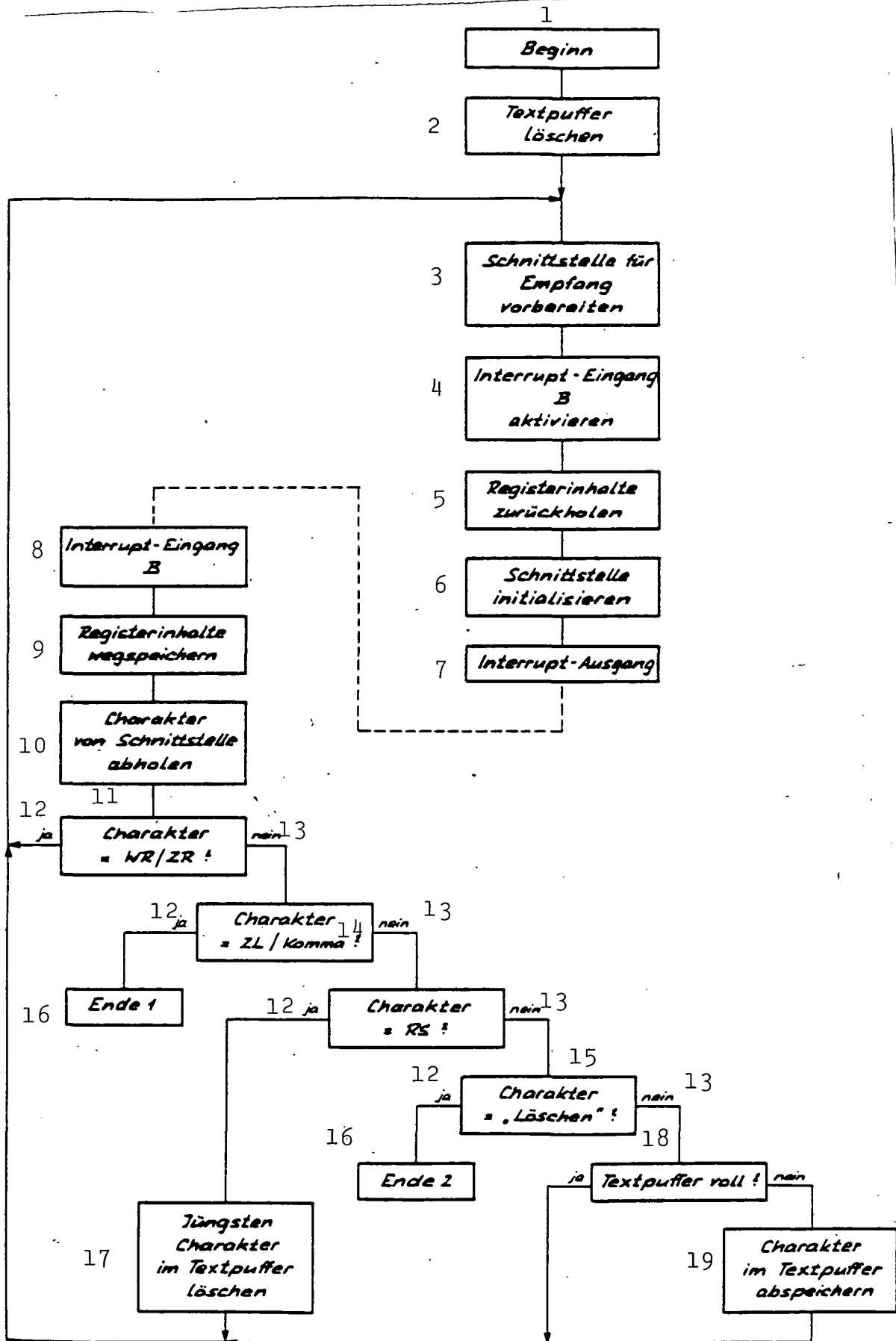
The driver routine C is also called by the dialogue routine. Its purpose is to give out answer sets to the display screen of the main terminal. The sets of answers must be obtained by the dialogue routine from a text buffer.

After the call the routine C prepares the interface to the main terminal for transmission and direct memory access and activates the interrupt input C. The duration from the initiation of the interface until the end of the text output is bridged over as an interrupt pause, during which functions at the lower priority levels of the program can have their chance.

The interrupt use is studied by routine C to find out whether it is caused by the orderly output or by intervention of the operator (pressing the break key). The result of the test is made known to the dialogue routine, which continues or breaks off its activity accordingly.

Protocol Routines D and E

These two driver routines start and end the output of protocol announcement on the display screen of the main terminal. The announcements are provided by the main program in a text buffer.



Dialogue Driver B of the Main Terminal

Key to Dialogue Driver B of the Main Terminal:

1. Beginning; 2. Erase text buffer; 3. Prepare interface for reception; 4. Activate interrupt input B; 5. Recover register contents; 6. Initiate interface; 7. Interrupt outlet; 8. Interrupt input B; 9. Store away register contents; 10. Collect symbols from interface; 11. Symbol; 12. Yes; 13. No; 14. Comma; 15. "Erase" symbol; 16. End; 17. Erase the latest symbol in the text buffer; 18. Is the text buffer full?; 19. Store symbol in text buffer.

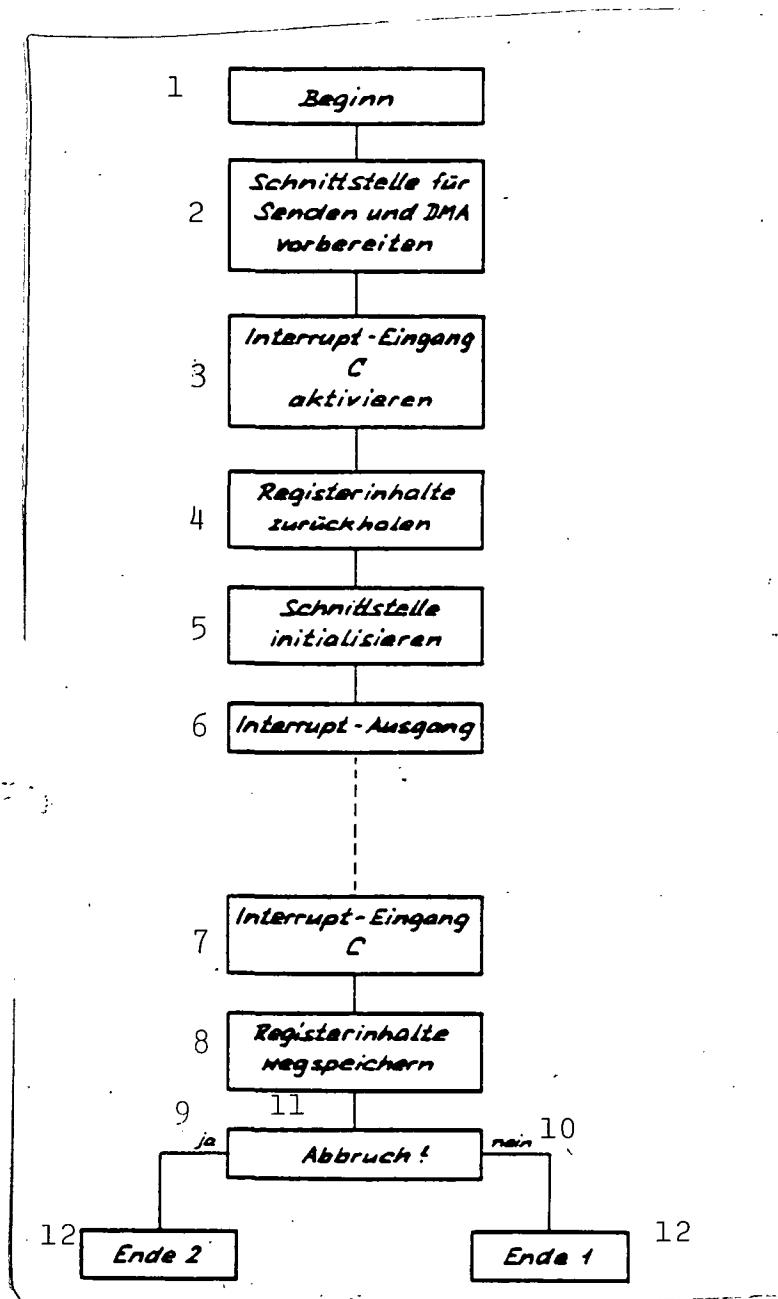
The driver D is called by the main program and first confirms whether the feeding protocol output has been ended in the meanwhile and released the interrupt E, or whether we have to wait for the end of a possible main terminal dialogue.

Subsequently routine D connects the interface for transmission and direct access to the memory, activates the interrupt input D and starts finally the interface for giving out the information.

/129

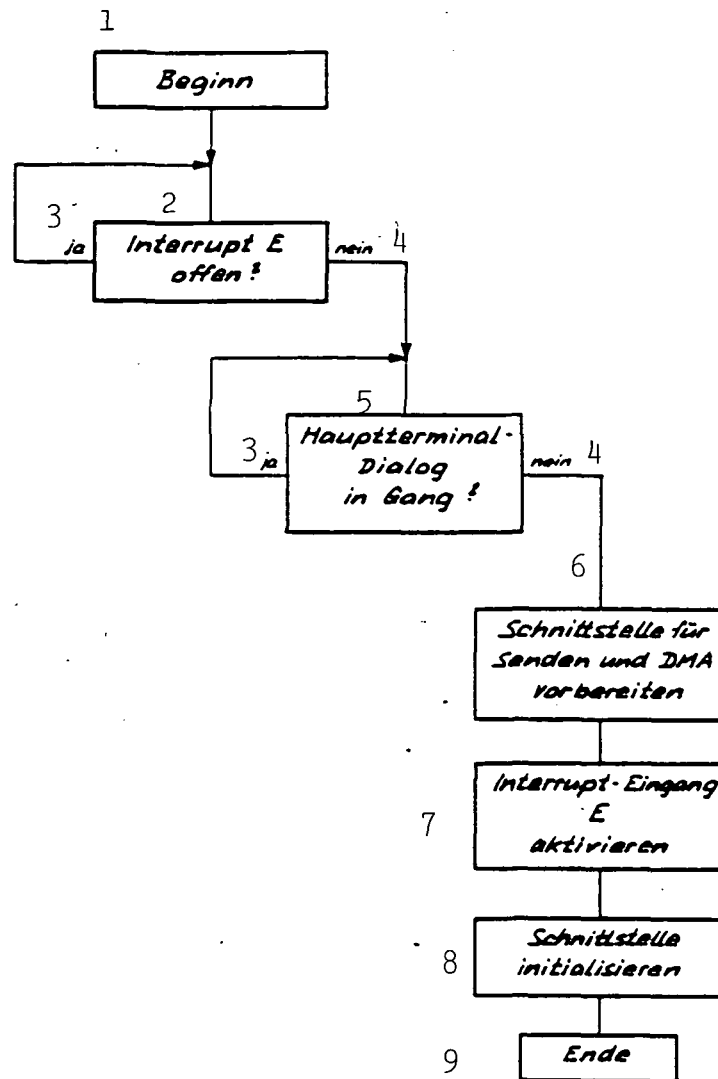
Driver routine E is called by the interrupt, which was caused either by the output of the last symbol from the text buffer, or by the operator by pressing the break key. Its purpose is to switch over the interface for reception, activation of the interrupt input A and starting the interface, providing the prerequisites for the reception of the dialogue by the operator.

If the interrupt E was released through the break key, this means that it is the operator's wish, by interrupting the current protocol announcement to intervene in the dialogue with the program control. Routine E takes this wish into account and it stops further protocol announcements by entering into a waiting loop, until the operator has intervened in the dialogue.



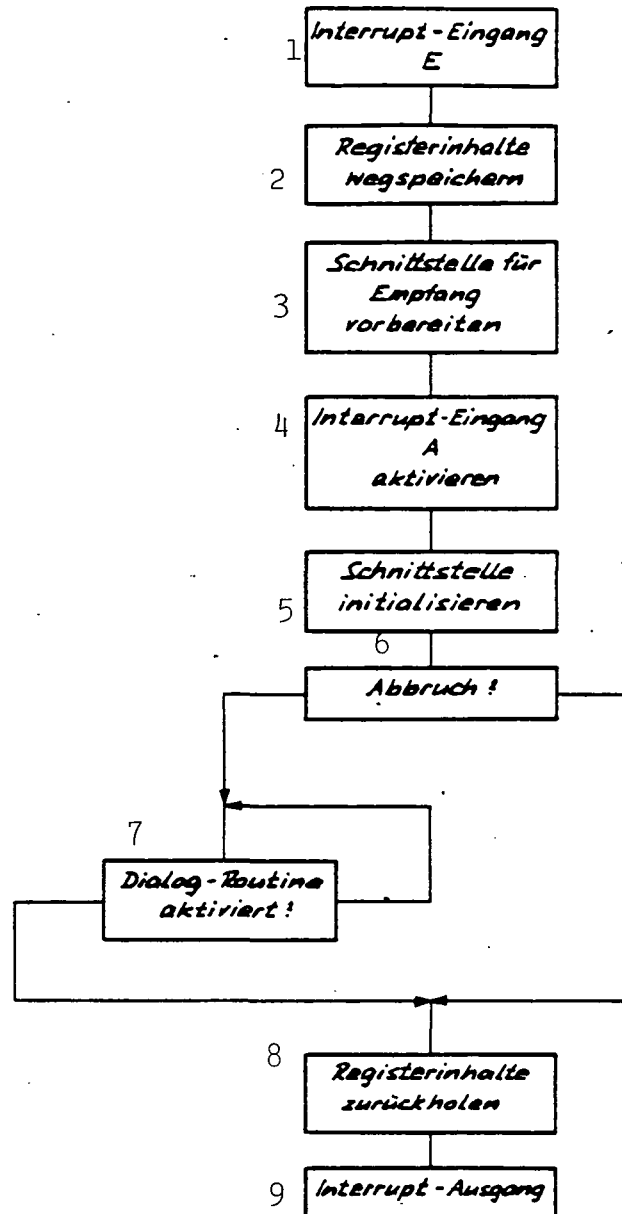
Dialogue Driver C of the Main Terminal

Key: 1. Beginning; 2. Prepare interface for transmission and DMA; 3. Activate interrupt input C; 4. Return register contents; 5. Initiate interface; 6. Interrupt outlet; 7. Interrupt input C; 8. Store away register contents; 9. Yes; 10. No; 11. Interruption?; 12. End



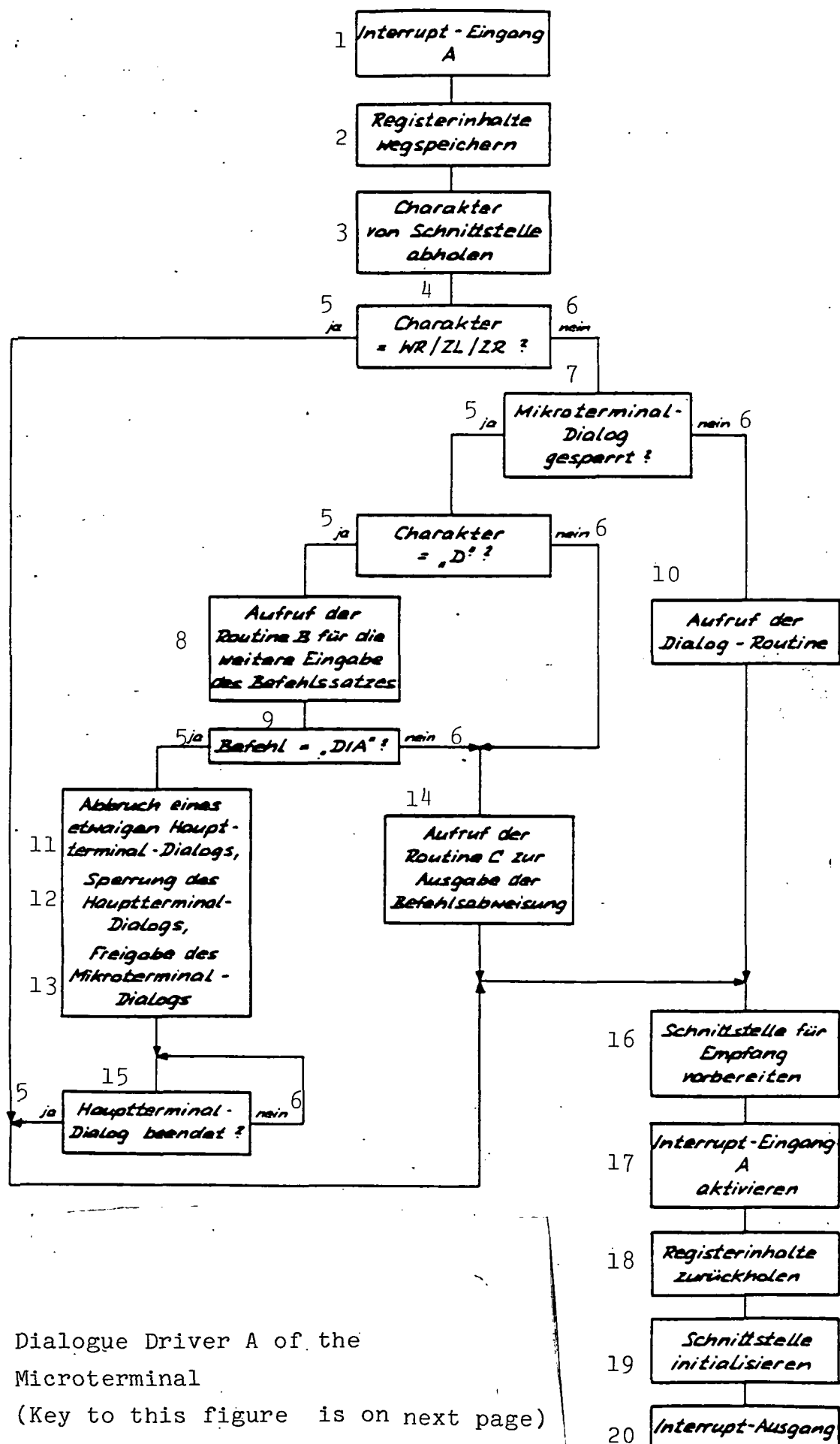
Protocol Driver D of the Main Terminal

Key: 1. Beginning; 2. Is the interrupt E open?
 3. Yes; 4. No; 5. Is the main terminal dialogue
 in progress?; 6. Prepare interface for transmission
 and DMA; 7. Activate interrupt input D;
 8. Initiate interface; 9. End



Protocol Driver E of the Main Terminal

Key: 1. Interrupt input E; 2. Store away register contents; 3. Prepare interface for reception; 4. Activate interrupt entrance A; 5. Initiate interface; 6. Is there interruption?; 7. Is the dialogue routine activated?; 8. Recover register contents; 9. Interrupt outlet



Dialogue Driver A of the
Microterminal
(Key to this figure is on next page)

Key to Dialogue Driver A of the Microterminal:

1. Interrupt input A; 2. Store away register contents;
3. Collect symbols from interface; 4. Symbol; 5. Yes; 6. No;
7. Is the microterminal dialogue blocked?;
8. Call of routine B for further input of order sets;
9. Order; 10. Call of dialogue routine;
11. Interruption of a possible main terminal dialogue;
12. Blocking of the main terminal dialogue;
13. Release of the microterminal dialogue;
14. Call of routine C for issuing order rejection;
15. Is the main terminal dialogue ended?;
16. Prepare interface for reception; 17. Activate interrupt input A;
18. Recover register contents; 19. Initiate interface;
20. Interrupt output

3.08 Driver for Microterminal

/132

The driver routines for the operation of the microterminal and its interface resemble substantially the routines A, B and C of the main terminal. In this connection both routines A are set up symmetrically because of the alternative blocking of the dialogue. Differences which may be expected because of the limited possibility for the microterminal, may be found and taken into account only in trial operation.

3.09 Driver for Experiment Computer

/134

The purpose of this driver is to communicate with the experiment-computer through a routine interface. The communication includes the transmission of a trigger symbol for program control and the subsequent transfer of a package of data in the opposite direction. The content of this data package describes the present direction and position of the gamma beam in the reference system of the experiment and provides information on the present operating state of the measurement table. By choosing the trigger symbol it is possible for the experiment computer to

release, besides the data transfer, also the movement of the measurement table to the next measurement position.

In the pauses between transmission, the driver routine waits with activated interrupt input A and interface switch ready for reception for the arrival of the trigger symbol. After its arrival the direction and position coordinates of the gamma beam and the status statement on the measurement table are transferred to an output buffer, that is according to the trigger symbol, released for the travel routine of the next movement stage. The transfer of the buffer content is started by the driver routine after programming the interface for transmission and direct access to memory, and activation of the interrupt input beam.

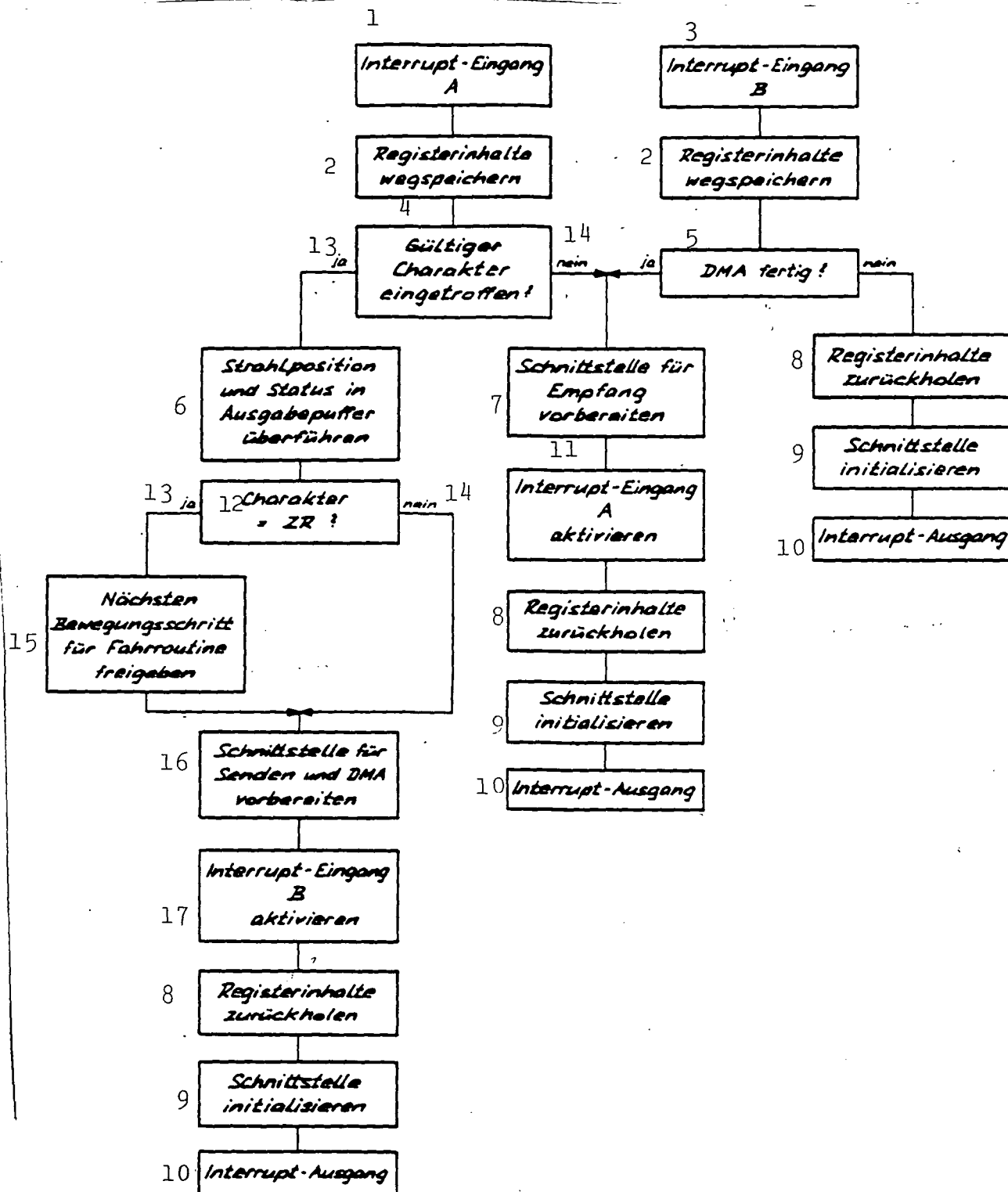
The next interrupt is produced by the interface when ending the data transfer and reaches the driver routine through input B. The latter is active only for a short time, to prepare everything for the arrival of the next trigger symbol by programming the interface for reception and activation of the interrupt input A.

3.10 Main Program

/136

The most important problems of the main program are:

- produce immediately after starting the program a definite basic state for all the functions of program and program control;
- switchover the type of operation by activation and de-activation of program functions;
- monitor during operation the status data coming from the other program routines and the function groups of program control;



Driver for the Experiment Computer

(Key to this figure is on following page)

Key to Driver for the Experiment Computer:

1. Interrupt input A; 2. Store away register contents;
3. Interrupt input B; 4. Has a valid symbol arrived?;
5. Is the DMA ready?; 6. Transfer beam position and status
to the output buffer; 7. Prepare interface for reception;
8. Recover register contents; 9. Initiate interface;
10. Interrupt outlet; 11. Activate interrupt input A;
12. Is the symbol "ZR"?; 13. Yes; 14. No; 15. Release next
movement stage for travel routine; 16. Prepare interface for
transmission and DMA; 17. Activate interrupt input beam

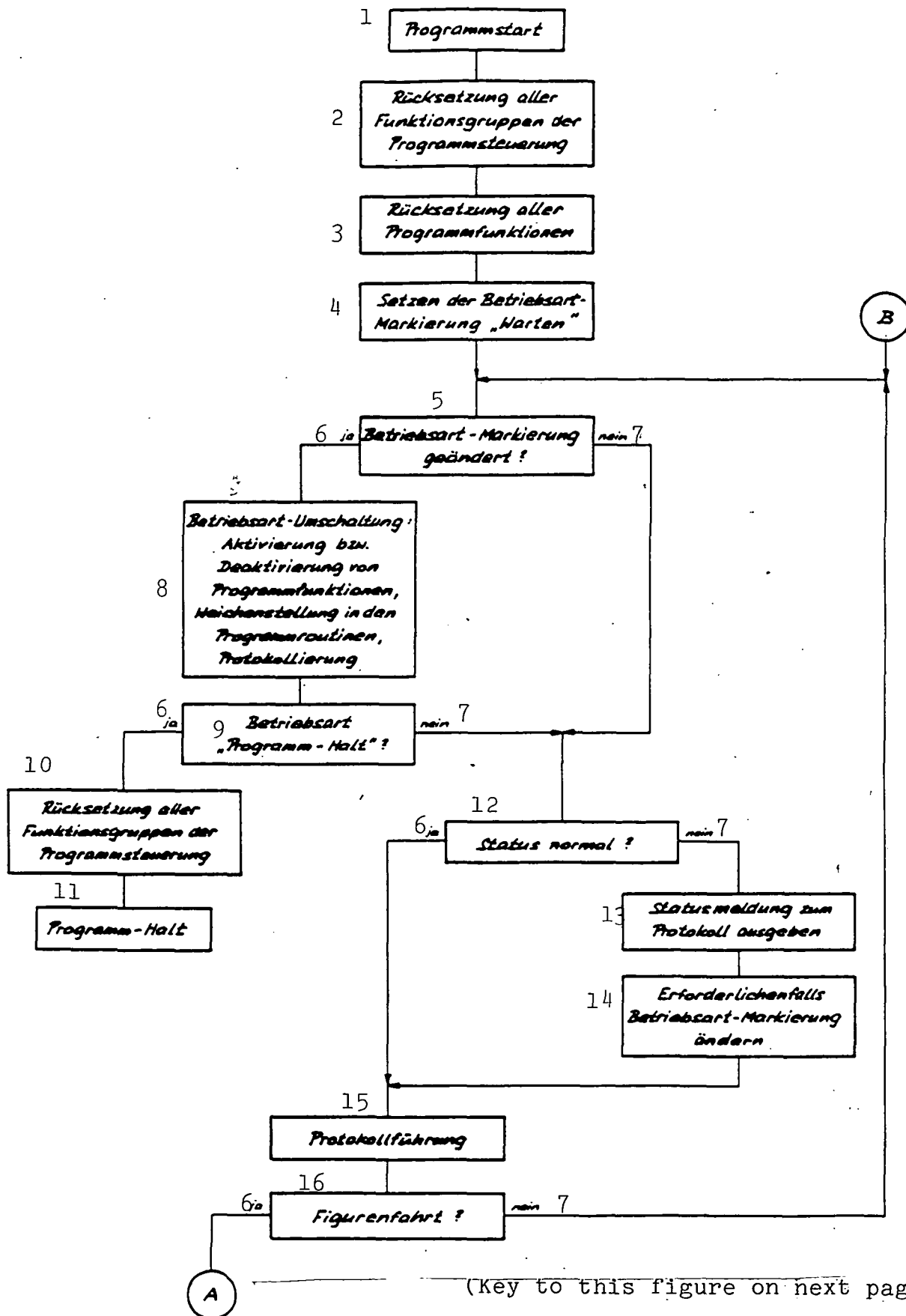
- give out also during operation status and protocol
announcements to the display screen of the main terminal;

- in types of operation with figure travel inspect the
figure travel plan according to the stored active figure sets
and their processing by calling the corresponding programmed
figure routine;

- after reaching the end of the figure travel plan,
automatic transition to the type of operation "waiting" or,
for figure travel after stacking, calling dialogue routine.

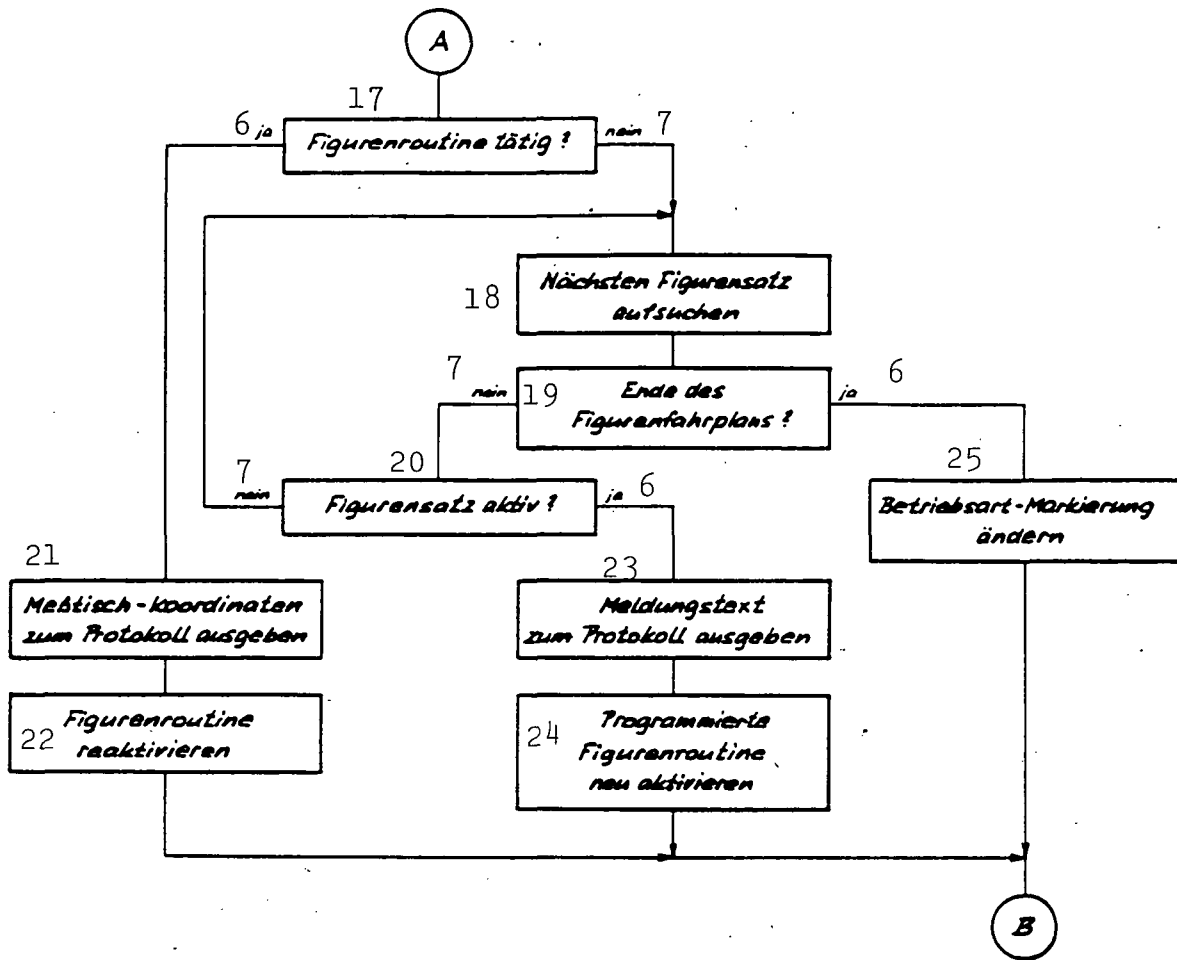
The main program assumes the topmost rank in the hierarchy
of program routines. It is the only partial program to be autonomous
and does not have the nature of a subroutine, is therefore called
neither by other program routines or by an interrupt. Rather
it always occupies the processor in the background of the program,
if there are no problems of higher priority at that time.

For this purpose the main program forms an operating loop
closed in itself, which is adjusted by soft adjustment to the
requirements of the existing type of operation. The above
indicated data are observed by program stations, which are
located all around the operating loop. The running time in
the loop increases naturally with the demands of the processor
by the sum of program activities.



(Key to this figure on next page)

Operating Loop of the Main Program



Operating Loop of the Main Program (Continued)

Key: 1. Program start; 2. Resetting of all the functional groups of the program control; 3. Resetting of all program functions; 4. Setting of the operating type marking "waiting"; 5. Is the operating time marking changed?; 6. Yes; 7. No; 8. Switchover of the type of operation, activation or deactivation of program functions, soft position in program routine, protocol establishment; 9. Is the type of operation "program halt?"; 10. Resetting of all functional groups of the program control; 11. Program halt; 12. Is the status normal?; 13. Status announcement for issuing the protocol; 14. If necessary change the marking for type of operations; 15. Protocol guiding; 16. Is it figure travel?; 17. Is the figure routine operational?; 18. Seek the next figure set; 19. Is it the end of the figure travel plan?; 20. Is the figure set active?; 21. Give out the measurement table coordinates to the protocol; 22. Reactivate figure routine; 23. Give out information text to the protocol; 24. Activate again programmed figure routines; 25. Change marking for type of operation